Séminaire Poincaré

Approaches to Quantum Error Correction

Julia Kempe
CNRS & LRI
Laboratoire de Recherche Informatique
Bât. 490, Université de Paris-Sud
91405 Orsay Cedex
France

We have persuasive evidence that a quantum computer would have extraordinary power. But will we ever be able to build and operate them?

A quantum computer will inevitably interact with its environment, resulting in decoherence and the decay of the quantum information stored in the device. It is the great technological (and theoretical) challenge to combat decoherence. And even if we can suitably isolate our quantum computer from its surroundings, errors in the quantum gates themselves will pose grave difficulties. Quantum gates (as opposed to classical gates) are unitary transformations chosen from a *continuous* set; they cannot be implemented with perfect accuracy and the effects of small imperfections in the gates will accumulate, leading to an eventual failure of the computation. Any reasonable correction-scheme must thus protect against small unitary errors in the quantum gates as well as against decoherence. Furthermore we must not ignore that the correction and recovery procedure itself can introduce new errors; successful fault-tolerant quantum computation must also deal with this issue.

The purpose of this account is to give an overview of the main approaches to quantum error correction. There exist several excellent reviews of the subject, which the interested reader may consult (see [Pre98b],[Pre99], [NC00], [KSV02], [Ste99, Ste01] and more recently [Got05]).

1 Introduction

"We have learned that it is possible to fight entanglement with entanglement."
John Preskill, 1996

In a ground breaking discovery in 1994, Shor [Sho94] has shown that quantum computers, if built, can factor numbers efficiently. Since then quantum computing has become a burgeoning field of research, attracting theoreticians and experimentalists alike, and regrouping researchers from fields like computer science, physics, mathematics and engineering. One more reason for the enormous impetus of this field is the fact that by the middle of 1996 it has been shown how to realize *fault-tolerant* quantum computation. This was not at all obvious; in fact it was not even clear how any form of quantum error-correction could work. Since then many new results about the power of quantum computing have been found, and the theory of quantum fault-tolerance has been developed and is still developing now.

In what follows we will give a simple description of the elements of quantum error-correction and quantum fault-tolerance. Our goal is to convey the necessary intuitions both for the problems and their solutions. In no way will we attempt to give the full and formal picture. This account is necessarily restricted with subjectively chosen examples and approaches and does not attempt to describe the whole field of quantum fault-tolerance, which has become a large subfield of quantum information theory of its own.

The structure of this account is the following. First we will describe why quantum error correction is a non-trivial achievement, given the nature of quantum information and quantum errors. Then we will briefly review the main features of a quantum computer, since, after all, this is the object we want to protect from errors, and it is also the object which will allow us to implement error-correction. Then we will give the first example of a quantum error-correcting code (the Shor-code), followed by other error correction mechanisms. We proceed to outline the elements of a full fledged fault-tolerant computation, which works error-free even though all of its components can be faulty. We mention alternative approaches to error-correction, so called error-avoiding or decoherence-free schemes. We finish with an outlook on the future. We will try to keep technical details and generalizations to a minimum; the interested reader will find more details, as well as suggestions for further reading, in the appendix.

2 The subtleties of quantum errors

"Small errors will accumulate and cause the computation to go off track."
Rolf Landauer, 1995, in "Is quantum mechanics useful?"

A quantum machine is far more susceptible to making errors then classical digital machines.

Not only is a quantum system more prone to decoherence resulting from unwanted interaction between the quantum system and its environment, but also, when manipulating quantum information we can only implement the desired transformation up to a certain precision. Until 1995 it was not clear at all if and how quantum error correction could work.

The second big breakthrough towards quantum computing (after Shor's algorithm) was the insight that quantum noise can be combatted or that quantum error protection and correction is possible. The first big step in this direction was again made by Peter Shor in his *"Scheme for reducing decoherence in a quantum memory"* in 1995 [Sho95].

This was indeed an amazing piece of work: the difficulties facing the introduction of classical error-correction ideas into the quantum realm seemed formidable. In fact there was a large number of reasons for pessimism. Let us cite but a few of the apparent obstacles:

(1) There is a hugely successful theory of classical error correction which allows to protect against classical errors. However, classical errors are *discrete* by their nature. In the most common case where the information is encoded into a string of bits, the possible errors are bit-flips or erasures. A quantum state is a priori continuous, and hence also the error is continuous. Similarly, quantum *operations* are continuous by their nature, and will necessarily only be implemented with a certain precision, but never exactly. As noted by Landauer [Lan95], small errors can accumulate over time and add up to large, uncorrectable errors. Moreover it is not clear how to adapt the discrete theory of error correcting codes to the quantum case.

(2) A second objection is that to protect against errors, the information must be encoded in a redundant way. However, the quantum no-cloning theorem [Die82, WZ82], which follows directly from the linearity of quantum mechanics, shows that it is impossible to copy an unknown quantum state. How then can the information be stored in a redundant way?

(3) Another point is the following: in order to correct an error, we need to first acquire some information about the nature and type of error. In other words we need to observe the quantum system, to perform a measurement. But any measurement collapses the quantum system and might destroy the information we have encoded in the quantum state. How then shall we extract information about the error without destroying the precious quantum superposition that contains the information?

Many researchers in the field were pessimistic about the prospects of error-correction and Shor's result came as a great surprise to many. All the initial objections let us appreciate the elegance of the solution even more. But before giving the key ingredients, we need to briefly review the object we want to protect form errors, the actual quantum machine.

3 What is a quantum computer?

"The more success the quantum theory has, the sillier it looks."
Albert Einstein, 1912

There are nearly as many proposals for the hardware of a quantum computer today, as there are experimental quantum physicists. The ultimate shape and function of a quantum computer will depend on the physical system used, be it optical lattices, large molecules, crystals or silicon based architectures. Nonetheless, each of these implementations have some key elements in common, since they all implement the quantum computing model.

What are the key ingredients of a quantum computer? A quantum computer is a machine that processes basic computational units, so called *qu*bits, quantum two-level systems. (Although there might be quantum machines which process higher-dimensional quantum systems, we will restrict ourselves for simplicity to the case of two-level systems.) Qubits are two-dimensional quantum states spanned by two basis-states, which we conventionally call $|0\rangle$ and $|1\rangle$, alluding to the classical bits of a standard computer. Hence the general state of a qubit is

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle \qquad |\alpha|^2 + |\beta|^2 = 1.$$

In each implementation of a quantum computer these basis states $|0\rangle$ and $|1\rangle$ need to be identified; they usually correspond to two chosen states of a larger system. For any quantum computation, *fresh* qubits have to be supplied in a known state, which is usually taken to be the $|0\rangle$ state.

A quantum computer implements a *unitary* transformation on the space of several qubits, as consistent with the laws of quantum mechanics. However, in the context of computation, each unitary is decomposed into *elementary* gates, where each gate acts on a small number of qubits only. These elementary gates constitute a universal gate set, which allows to implement any unitary operation on the set of qubits. There are several universal gate sets, but we will mention only two, which are relevant for what follows. The first such set is continuous, and consists of all one-qubit unitaries, together with the controlled NOT or CNOT. The action of the CNOT on the basis states $|00\rangle, |01\rangle, |10\rangle, |11\rangle$ is as follows

$$CNOT = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}.$$

In quantum circuit design it is often depicted as in Fig. 1.

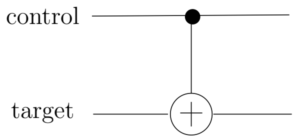

Figure 1: The top qubit is the *control* qubit. If it is in the state $|0\rangle$, then the *target* qubit stays unchanged, if it is in the state $|1\rangle$, the target qubit is flipped from $|0\rangle$ to $|1\rangle$ and vice versa.

It is possible to implement any unitary operation by a sequence of CNOT and single qubit unitary operations on the qubits.

The second set of universal gates is *discrete*. It contains the gates H, $\pi/8$, Z and $CNOT$. The first three gates are single qubit gates. H is called the *Hadamard* transform, $\pi/8$ is a phase gate and Z is known to physicists as one of the Pauli matrices σ_z. On the basis $|0\rangle$, $|1\rangle$, they act as

$$H = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \qquad \pi/8 = \begin{pmatrix} e^{i\frac{\pi}{8}} & 0 \\ 0 & e^{-i\frac{\pi}{8}} \end{pmatrix} \qquad Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}.$$

All experimental proposals, in one way or another, demonstrate the ability to induce the transformations corresponding to this (or some other universal) gate set. Note that it is absolutely crucial to implement the two-qubit gate CNOT (or some other suitable two-qubit gate), as single qubit operations alone are clearly not universal.

This gate set is discrete, it contains only four gates. However, this comes at a price. It is not in general possible to implement any unitary transformation with a sequence drawn from this gate set. But it is possible to *approximate* any unitary to arbitrary accuracy using gates from this set. (Here accuracy is measured as the spectral norm of the difference between the desired unitary matrix and the actually implemented unitary matrix.) This is good enough for our purposes.

The last ingredient of a quantum computer is the *read-out*, or measurement. At the end of the day, when we want to extract the result of the quantum computation, we need to observe the quantum system, to gain information about the result.

In general one assumes that each qubit (or the qubit carrying the result of the computation) is measured in some basis. The classical result represents the outcome of the computation.

Schematically, then, a quantum computer looks like in Fig. 2.

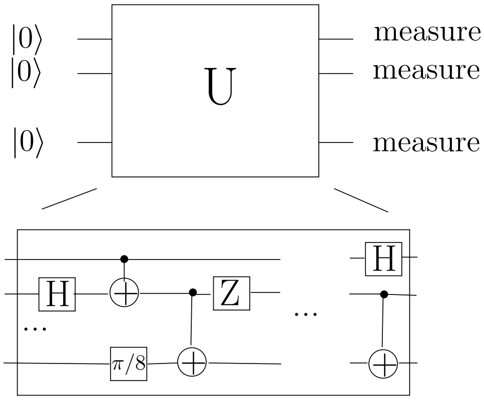

Figure 2: A quantum computer, schematically. Fresh qubits, initialized in the state $|0\rangle$, are supplied as the input to the unitary transform U. U is composed of elementary gates affecting at most 2 qubits. At the end of the computation the qubits are measured.

4 What is a quantum error?

"Had I known that we were not going to get rid of this damned quantum jumping, I never would have involved myself in this business!"
Erwin Schrödinger

Quantum computers are notoriously susceptible to quantum errors, and this is certainly the reason we did not yet succeed in building a scalable model. The problem is that our quantum system is inevitably in contact with a larger system, its *environment*. Even if we make heroic efforts to isolate a quantum system form its environment, we still have to manipulate the information inside it in order to compute, which again will introduce errors. Where it not for the development of methods of quantum error correction, the prospects for quantum computing technology would not be bright. In order to describe quantum error correction we need to get a clear picture of what the noise processes affecting our machine are.

But how do we describe a quantum error?

Let us study the example of a single qubit. This qubit might undergo some random unitary transformation, or it might decohere by becoming entangled with the environment. In general it will undergo some unitary transformation in the *combined* system of qubit and environment. Let us call $|E\rangle$ the state of the environment before the interaction with the qubit. Then the most general unitary transformation on system and environment can be described as

$$\begin{aligned} U: \quad & |0\rangle \otimes |E\rangle \longrightarrow |0\rangle \otimes |E_{00}\rangle + |1\rangle \otimes |E_{01}\rangle \\ & |1\rangle \otimes |E\rangle \longrightarrow |0\rangle \otimes |E_{10}\rangle + |1\rangle \otimes |E_{11}\rangle. \end{aligned}$$

Here $|E_{ij}\rangle$ represent not necessarily orthogonal or normalized states of the environment, with the only constraint that the total evolution be unitary. The unitary U *entangles* our qubit with the environment. Potentially, this entanglement will lead to decoherence of the information stored in the qubit.

Suppose now the qubit is in the state $\alpha|0\rangle + \beta|1\rangle$[1]. Now if the qubit is afflicted by an error, it evolves as

$$\begin{aligned}
(\alpha|0\rangle + \beta|1\rangle) \otimes |E\rangle \quad \longrightarrow \quad & \alpha\left(|0\rangle \otimes |E_{00}\rangle + |1\rangle \otimes |E_{01}\rangle\right) + \beta\left(|0\rangle \otimes |E_{10}\rangle + |1\rangle \otimes |E_{11}\rangle\right) \\
& = (\alpha|0\rangle + \beta|1\rangle) \otimes \frac{1}{2}\left(|E_{00}\rangle + |E_{11}\rangle\right) \qquad \textit{identity} \\
& + (\alpha|0\rangle - \beta|1\rangle) \otimes \frac{1}{2}\left(|E_{00}\rangle - |E_{11}\rangle\right) \qquad \textit{phase flip} \\
& + (\alpha|1\rangle + \beta|0\rangle) \otimes \frac{1}{2}\left(|E_{01}\rangle + |E_{10}\rangle\right) \qquad \textit{bit flip} \\
& + (\alpha|1\rangle - \beta|0\rangle) \otimes \frac{1}{2}\left(|E_{01}\rangle - |E_{10}\rangle\right) \qquad \textit{bit \& phase flip.} \qquad (1)
\end{aligned}$$

Intuitively, we may interpret this expansion by saying that one of four things happens to the qubit: nothing, a bit flip, a phase flip or a combination of bit flip and phase flip. This will be made more precise in the next section, where we see that quantum error correction will include a *measurement* of the error, collapsing the state into one of the four possibilities above. This way, even though the quantum error is continuous, it will become *discrete* in the process of quantum error correction. We will denote the four errors as acting on a qubit as

$$\underbrace{I = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}}_{\textit{identity}} \qquad \underbrace{X = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}}_{\textit{phase flip}} \qquad \underbrace{Z = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}}_{\textit{bit flip}} \qquad \underbrace{Y = XZ = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}}_{\textit{bit \& phase flip}}. \qquad (2)$$

These four matrices form the so called *Pauli group*. Another way of saying the above is to realize that these four errors span the space of unitary matrices on one qubit, i.e. any matrix can be expressed as a linear combination of these four matrices (with complex coefficients). If we trace out the environment (average over its degrees of freedom, see App. B.2), the resulting operator can be expanded in terms of the Pauli group, we can attach a probability to each Pauli group element. Often the analysis of fault-tolerant architectures is simplified by assuming that the error is a random non-identity Pauli matrix with equal probability $\varepsilon/3$, where ε is the *error rate*.

We now make the crucial assumption: that the error processes affecting different qubits are *independent* from each other. A quantum error correcting code, then, will be such that it can protect against these four possible errors. Once the error has become discrete it is much more obvious how to apply and extend classical error correction codes, which are able to protect information against a bit flip.

We have so far only analyzed errors due to decoherence, but have neglected errors due to imperfections in the gates, in the measurement process and in preparation of the initial states. All these operations can be faulty. A natural assumption is again that these imperfections are independent of each other. In a similar fashion as before we can discretize the errors in a quantum gate. We can model a faulty gate by assuming that is is a perfect gate, followed by an error. For a one-qubit gate this error is the same as given in Eq. (1). For a two-qubit gate we assume that both qubits undergo possibly correlated decoherence. Similar reasoning as in Eq. (1) shows, that in that case the error is a linear combination of 16 possible errors, resulting from all combinations of the errors in Eq. (2) on both qubits. Again, often the additional assumption is made that all 15 non-identity errors appear with equal probability $\varepsilon_2/15$, where ε_2 is the two-qubit gate error rate. In a similar fashion we will deal with measurement and state preparation errors.

Note that our analysis of the error is somewhat simplified. Several tools have been developed to study quantum decoherence and quantum noise. Some of these formalisms are described in more detail in App. B. As already mentioned, in order to give methods for quantum error correction, some assumptions about the nature of the noise have to be made. In one of the common models of noise in a quantum register it is assumed that each qubit interacts *independently* with the

[1] Of course our qubit could be part of a larger quantum state of several qubits. It might be entangled with other qubits which are unaffected by errors. So the coefficients α and β need not be numbers, they can be states that are orthogonal to both $|0\rangle$ and $|1\rangle$.

environment in a Markovian fashion[2]; the resulting errors are single qubit errors affecting each qubit independently at random. More details on models of quantum noise are given in App. C.

5 The first error correction mechanisms

"Correct a flip and phase - that will suffice.
If in our code another error's bred,
We simply measure it, then God plays dice,
Collapsing it to X or Y or Zed."
Daniel Gottesman, in "Error Correction Sonnet"

We have seen how entanglement with the environment can cause errors that result in a complete loss of the quantum information. However, entanglement will also allow us to *protect* the information in a non-local way. If we distribute the information over several qubits in a way that it cannot be accessed by measuring just a few of the qubits, then by the same token it cannot be damaged if the environment interacts with just a few of the qubits.

A marvelous machinery has been developed in the classical world to protect classical information, the theory of error correcting codes. The simplest possible such code is the *repetition* code: each bit is replaced by three of its copies:

$$\mathcal{C}: \quad 0 \longrightarrow 000 \qquad 1 \longrightarrow 111.$$

This code clearly protects against one bit flip error. If a bit is flipped, we can still decode the information by majority voting. Only if two bit flips happen we will be unable to correctly decode the information. But if we assume that the probability of a bit flip is ε and independent on each bit, then the probability that we cannot correct a bit flip is $3\varepsilon^2(1-\varepsilon)+\varepsilon^3$ (there are three possible ways to have two bit flips and one way to have three bit flips). If we would not encode the information at all the error probability is ε, so as long as $\varepsilon < 1/2$ we gain by encoding.

But how can we extend this idea to the quantum setting? There is no way to copy quantum information. There are not only bit flip, but also phase flip errors (and combinations of both). And moreover a measurement for majority vote will cause disturbance.

Shor was the first to overcome all these obstacles [Sho95]. He gave the most straightforward quantum generalization of the repetition code. Suppose we want to just deal with bit flip errors. We encode a single qubit with the repetition code on the basis states, i.e.

$$|0\rangle \longrightarrow |000\rangle \qquad |1\rangle \longrightarrow |111\rangle,$$

such that

$$\alpha|0\rangle + \beta|1\rangle \longrightarrow \alpha|000\rangle + \beta|111\rangle. \tag{3}$$

This encoding can be realized with the circuit in Fig. 3.

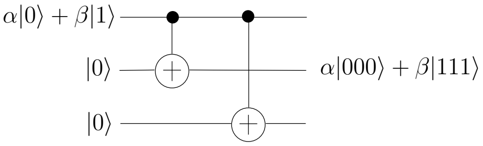
$\alpha|0\rangle + \beta|1\rangle$
$|0\rangle$
$|0\rangle$
$\alpha|000\rangle + \beta|111\rangle$

Figure 3: The CNOTs flip the target qubit if the first qubit is in the state $|1\rangle$. Note that the transformation does not *copy* the state of the first qubit to the other two qubits, but rather implements the transformation of Eq.(3).

Now suppose a bit flip happens, say on the first qubit. The state becomes $\alpha|100\rangle + \beta|011\rangle$. If we measured the qubits in the computational basis, we would obtain one of the states $|100\rangle$ or $|011\rangle$, but we would destroy the quantum superposition. But what if instead we measured the *parity* of all pairs of qubits, without acquiring any additional information? For instance we can measure the parity of the first two qubits with the circuit in Fig. 4.

[2] This means that the environment maintains no memory of the errors, which are thus *uncorrelated* in *time* and qubit *location*.

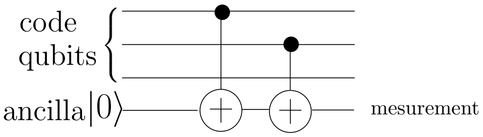

Figure 4: Circuit to measure the parity of the first two qubits of the quantum code word. Each CNOT flips the ancilla qubit if the source qubit is in the state $|1\rangle$. If the first two qubits are in the state $|00\rangle$, the ancilla is left in the state $|0\rangle$. If these qubits are in the state $|11\rangle$ the ancilla is flipped twice and its state is also $|0\rangle$. Otherwise it is flipped once by one of the CNOTs.

In our example, a parity measurement does not destroy the superposition. If the first qubit is flipped, then both $|100\rangle$ and $|011\rangle$ have the same parity 1 on the first two qubits. If no qubit is flipped and the code word is still in the state of Eq. (3) this parity will be 0 for both $|000\rangle$ and $|111\rangle$. If the error is a *linear combination* of identity and bit flip, similar to Eq. (1), then the measurement will *collapse* the state into one of the two cases. Let us adapt Eq. (1) to the case of only a bit flip error on one qubit ($|E_{00}\rangle = |E_{11}\rangle$, $|E_{01}\rangle = |E_{10}\rangle$ and $|E_{01}\rangle$ and $|E_{00}\rangle$ are orthogonal) and write

$$(\alpha|0\rangle + \beta|1\rangle) \otimes |E\rangle \quad \longrightarrow \quad \sqrt{1-\varepsilon}\,\underbrace{(\alpha|0\rangle + \beta|1\rangle)}_{identity} \otimes |\widetilde{E}_{00}\rangle + \sqrt{\varepsilon}\,\underbrace{(\alpha|1\rangle + \beta|0\rangle)}_{bit\,flip} \otimes |\widetilde{E}_{01}\rangle, \tag{4}$$

where we have normalized the state of the environment ($|\widetilde{E}_{00}\rangle$ and $|\widetilde{E}_{01}\rangle$ have norm 1)[3]. The probability that the parity measurement collapses to the bit flip case is ε, the probability to project onto a state where no error has happened is $1-\varepsilon$. Imagine now that each of the three qubits of the code undergoes the same error process of Eq. (4). This gives a threefold tensor product of Eq. (4) (each qubit has its own environment state), which shows that the probability of no error becomes $(1-\varepsilon)^3 \geq 1-3\varepsilon$, and the probability of each of the single qubit errors is $\varepsilon(1-\varepsilon) < \varepsilon$. Of course there is now a nonzero probability that the state will be collapsed to a state where two or even three single qubit errors occurred; however, the total probability of this happening is given by $3\varepsilon^2(1-\varepsilon) + \varepsilon^3 \leq 3\varepsilon^2$.

This mechanism illustrates how a measurement that detects the error, also discretizes it. The parity measurement *disentangles* the code qubits from the environment and acquires information about the error. The three parities (for each qubit pair of the code word) give complete information about the location of the bit flip error. They constitute what is called the error *syndrome* measurement. The syndrome measurement does not acquire any information about the encoded superposition, and hence it does not destroy it. Depending on the outcome of the syndrome measurement, we can correct the error by applying a bit flip to the appropriate qubit.

We have successfully resolved the introduction of redundancy, the discretization of errors and a way to measure the syndrome without destroying the information. We still need to take care of phase flip errors. We have been able to protect against bit flip errors by encoding the bits redundantly. The idea is to also encode the phase of the state in a redundant fashion. Shor's idea was to encode a qubit using *nine* qubits in the following way:

$$\begin{aligned}
|0\rangle_{enc} &= \frac{1}{\sqrt{2^3}}\,(|000\rangle + |111\rangle)\,(|000\rangle + |111\rangle)\,(|000\rangle + |111\rangle) \\
|1\rangle_{enc} &= \frac{1}{\sqrt{2^3}}\,(|000\rangle - |111\rangle)\,(|000\rangle - |111\rangle)\,(|000\rangle - |111\rangle)
\end{aligned} \tag{5}$$

Note that with this encoding, each of the blocks of three qubits is still encoded with a repetition code, so we can still correct bit flip errors in a fashion very similar to above. But what about phase

[3] Note, that if we trace out the environment (see App. B.2), we obtain a process where with probability $1-\varepsilon$ nothing happens, and with probability ε the bit is flipped. ε defines the *rate* of error.

errors? A phase flip error, say on one of the first three qubits, acts as:

$$|0\rangle_{enc} \overset{phase\,flip}{\longrightarrow} \frac{1}{\sqrt{2^3}} \left(|000\rangle - |111\rangle\right) \left(|000\rangle + |111\rangle\right) \left(|000\rangle + |111\rangle\right)$$
$$|1\rangle_{enc} \overset{phase\,flip}{\longrightarrow} \frac{1}{\sqrt{2^3}} \left(|000\rangle + |111\rangle\right) \left(|000\rangle - |111\rangle\right) \left(|000\rangle - |111\rangle\right)$$

We need to detect this phase flip *without* measuring the information in the state. To achieve this we will follow the ideas developed for the bit flip and measure the parity of the phases on each pair of two of the three blocks. There is an interesting and useful duality between bit flip and phase flip errors. Let us look at a different basis for qubits, given by the states

$$|+\rangle = \frac{1}{\sqrt{2}} \left(|0\rangle + |1\rangle\right) \qquad |-\rangle = \frac{1}{\sqrt{2}} \left(|0\rangle - |1\rangle\right)$$

The change from the standard basis to the $|\pm\rangle$-basis we apply the Hadamard transform H. Now note that a phase flip error acts as

$$|+\rangle \overset{phase\,flip}{\longrightarrow} |-\rangle \qquad |-\rangle \overset{phase\,flip}{\longrightarrow} |+\rangle. \tag{6}$$

In other words a phase flip in the standard basis becomes a *bit flip* in the $|\pm\rangle$-basis. If we apply a Hadamard transform to each of the three qubits of a block of the Shor code, we obtain

$$H^{\otimes 3} \frac{1}{\sqrt{2}} \left(|000\rangle + |111\rangle\right) = \frac{1}{2} \left(|000\rangle + |110\rangle + |101\rangle + |011\rangle\right)$$
$$H^{\otimes 3} \frac{1}{\sqrt{2}} \left(|000\rangle - |111\rangle\right) = \frac{1}{2} \left(|111\rangle + |001\rangle + |010\rangle + |100\rangle\right)$$

Note that the parity of each of the bitstrings for positive phase is *even* and for negative phase it is *odd*. We can see that if two blocks have different phase, then in the parity of its constituent 6 qubits is odd, otherwise it is even. Hence, in order to detect a phase error, we just need to measure the parity of all qubits in the three possible pairs of blocks in the $|\pm\rangle$ basis. The circuit in Fig. 5 does exactly that.

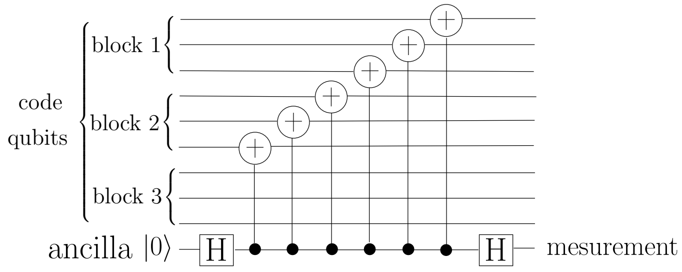

Figure 5: Circuit to measure the parity of the *phase* of the first and the second block of three qubits. In the $|\pm\rangle$-basis a CNOT acts on target (t) and control (c) bit as $|+\rangle_t|\pm\rangle_c \rightarrow |+\rangle_t|\pm\rangle_c$ and $|-\rangle_t|\pm\rangle_c \rightarrow |-\rangle_t|\mp\rangle_c$, i.e. it flips the *control* bit in the $|\pm\rangle$ basis if the *target* bit is $|-\rangle_c$. This way the ancilla bit is flipped an even number of times from $|+\rangle$ to $|-\rangle$ if blocks 1 and 2 have the same phase, and an odd number of times if they have different phase.

The nine-bit Shor code above protects against bit and phase flip, and also against a combination of both (when *both* bit and phase flip are detected, the error is XZ). Note again, that we assume that *each* of the qubits undergoes some error at rate ε. Hence, by the discretization resulting from the error-recovery measurement, the state will be projected onto either a state where no error has occurred (with probability $\geq 1 - 9\varepsilon$) or a state with a large error (single qubit, two qubit etc.). This code protects against all single qubit errors. Only when two or more errors occur (which happens with probability $\leq 36\varepsilon^2$) the error is irrecoverable. Comparing this with the error rate of an unencoded qubit, ε, we see that this code is advantageous whenever $\varepsilon \leq 1/36$.

6 Quantum Error Correcting Codes

"If people do not believe that mathematics is simple, it is only because they do not realize how complicated life is."
John von Neumann

Let us internalize the crucial properties of Shor's code: A small part of the Hilbert space of the system is designated as the *code subspace* $\mathcal{C}$. In the Shor code $\mathcal{C}$ is spanned by the two states in Eq. (5). We have a discrete set of correctable errors $\{\mathbf{E}_\alpha\}$. Each of the correctable errors $\mathbf{E}_\alpha$ maps the code space $\mathcal{C}$ to a mutually *orthogonal* error space. We can make a measurement that tells us in which of the mutually orthogonal spaces the system resides, and hence exactly infer the error. The error can be repaired by applying an appropriate unitary transformation ($\mathbf{E}_\alpha^\dagger$).

These ideas have been formalized to define *quantum error correcting codes (QECCs)*. An (N, K) quantum error correcting code $\mathcal{C}$ is a K dimensional subspace of an N dimensional Hilbert space (coding space $\mathcal{H}$) together with a recovery (super)operator $\mathcal{R}$. The recovery operator usually consists of some sort of measurement (to detect the error) followed by a conditional unitary to correct it, but we do not necessarily have to think about it in this way. The code $\mathcal{C}$ is *$\mathcal{E}$-correcting* if on the code-space an error followed by recovery restores the codeword, i.e.

$$\mathcal{R} \circ \mathcal{E} = \mathcal{I} \quad \text{on} \quad \mathcal{C}$$

It has been shown [BDSW96, KL97] that QECC's exist for the set of errors if the following conditions (*QECC-conditions*) are satisfied:

QECC-conditions: Let $\mathbf{E}$ be a discrete linear base set for $\mathcal{E}$ and let the code $\mathcal{C}$ be spanned by the basis $\{|\Psi_i\rangle : i = 1 \dots K\}$. Then $\mathcal{C}$ is an $\mathcal{E}$-correcting QECC if and only if $\forall |\Psi_i\rangle, |\Psi_j\rangle \in \mathcal{C}$

$$\langle \Psi_j | \mathbf{E}_\beta^\dagger \mathbf{E}_\alpha | \Psi_i \rangle = c_{\alpha\beta} \delta_{ij} \quad \forall \mathbf{E}_\alpha, \mathbf{E}_\beta \in \mathbf{E}. \tag{7}$$

What this means is the following: Errors $\mathbf{E}_\alpha, \mathbf{E}_\beta \in \mathbf{E}$ acting on *different* orthogonal codewords $|\Psi_i\rangle$ take these codewords to *orthogonal* states ($\langle \Psi_i | \mathbf{E}_\beta^\dagger \mathbf{E}_\alpha | \Psi_j \rangle = 0$). Otherwise errors would destroy the perfect distinguishability of orthogonal codewords and no recovery would be possible. On the other hand for different errors acting on the *same* codeword $|\Psi_i\rangle$ we only require that $\langle \Psi_i | \mathbf{E}_\beta^\dagger \mathbf{E}_\alpha | \Psi_i \rangle$ does not depend on i. Otherwise we would - in identifying the error - acquire some information about the encoded state $|\Psi_i\rangle$ and thus inevitably disturb it.

We usually think of the errors $\mathbf{E}_\alpha$ to be a subset of the Pauli group with up to t non-identity Pauli matrices (for a t-error correcting QECC).

It is now possible to make the connection to the theory of classical error correcting codes. It turns out that there are families of classical codes with certain properties (concerning their dual) which make good quantum error correcting codes [Ste96b, CS96]. The codes have become known as Calderbank-Shor-Steane codes (CSS codes). It has been shown that for any number t of correctable errors, there is a QECCs which can correct up to t errors (bit flip, phase flip and combination). As a result this code reduces the error for an unencoded qubit, ε, to $c\varepsilon^{t+1}$, where c is a constant depending on the code.

To illustrate this connection to classical codes we will briefly describe the smallest code in that family, which was first given by Steane [Ste96b]. This is the so called 7 qubit *Steane code*, based on the classical 7-bit Hamming code. The classical Hamming code encodes one bit into 7 bits. The codewords can be characterized by the *parity check* matrix

$$H = \begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{pmatrix}. \tag{8}$$

The code is the kernel of H, i.e. each code word is a 7-bit vector v_{code} such that $H \cdot v_{code} = (0, 0, 0)^T$ in $GF(2)$ arithmetic. H has three linearly independent rows (over $GF(2)$), so the kernel is spanned by four linearly independent code words, and hence there are 16 different code words. If an error

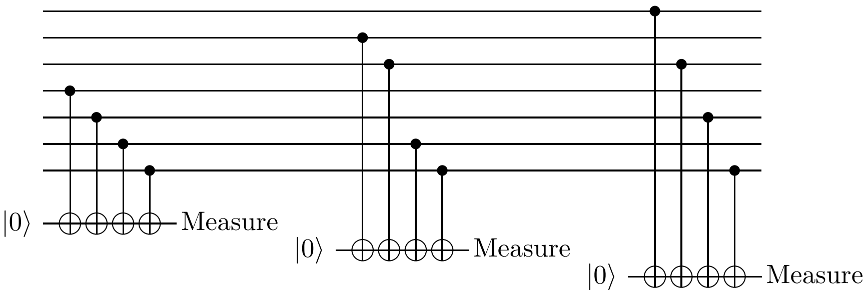

Figure 6: Computation of the bit-flip syndrome for Steane's 7-qubit code. The three ancilla qubits carry the error syndrome.

affects the ith bit of the codeword, this codeword is changed to $v_{code}+e_i$. The parity check matrix of the resulting word is $H(v_{code}+e_i)=He_i\neq 0$, which is just the ith column of H. Since all columns of H are distinct, each e_i has a different *error syndrome* and we can infer e_i from it.

Steane's code, derived from the Hamming code, is the following

$$\begin{aligned}|0\rangle_{code} = \tfrac{1}{\sqrt{8}}\Big(\textstyle\sum_{\substack{\text{even } v\\ \in \text{ Hamming}}}|v\rangle\Big) = \tfrac{1}{\sqrt{8}} & \Big(|0000000\rangle+|0001111\rangle+|0110011\rangle+|0111100\rangle \\ & +|1010101\rangle+|1011010\rangle+|1100110\rangle+|1101001\rangle\Big)\ , \\ |1\rangle_{code} = \tfrac{1}{\sqrt{8}}\Big(\textstyle\sum_{\substack{\text{odd } v\\ \in \text{ Hamming}}}|v\rangle\Big) = \tfrac{1}{\sqrt{8}} & \Big(|1111111\rangle+|1110000\rangle+|1001100\rangle+|1000011\rangle \\ & +|0101010\rangle+|0100101\rangle+|0011001\rangle+|0010110\rangle\Big),\end{aligned} \tag{9}$$

i.e. $|0\rangle_{code}$ is the superposition of all even and $|1\rangle_{code}$ the superposition of all odd codewords. Note that all states appearing in the code words are Hamming code words, and hence a single bit flip can be detected by a simple parity measurement, as in Fig. 6.

To deal with phase flip errors we use the observation of Eq. (6), that phase flip errors correspond to bit flip errors in the $|\pm\rangle$ basis. But if we change to this basis by applying the Hadamard transform to each bit, we obtain

$$\begin{aligned}H^{\otimes 7}|0\rangle_{\text{code}} = & \quad \tfrac{1}{4}\Big(\textstyle\sum_{\substack{v\in\\ \text{Hamming}}}|v\rangle\Big) = \tfrac{1}{\sqrt{2}}\left(|0\rangle_{\text{code}}+|1\rangle_{\text{code}}\right)\ , \\ H^{\otimes 7}|1\rangle_{\text{code}} = & \ \tfrac{1}{4}\Big(\textstyle\sum_{\substack{v\in\\ \text{Hamming}}}(-1)^{wt(v)}|v\rangle\Big) = \tfrac{1}{\sqrt{2}}\left(|0\rangle_{\text{code}}-|1\rangle_{\text{code}}\right),\end{aligned} \tag{10}$$

(where $wt(v)$ denotes the weight of v). The key point is that in the $|\pm\rangle$ basis, like in the $|0\rangle,|1\rangle$ basis, $|0\rangle_{code}$ and $|1\rangle_{code}$, are superpositions of Hamming codewords. Hence, in the rotated basis, as in the original basis, we can perform the Hamming parity check to diagnose bit flips, which are phase flips in the original basis. Assuming that only one qubit is in error, performing the parity check in both bases completely diagnoses the error, and enables us to correct it.

The core observation that allows to generalize Steane's construction to codes that encode more bits and can correct more errors is the following: If a quantum code word is a linear superposition over classical code words that form a code $\mathcal{C}$, then in the $|\pm\rangle$ basis this code word is a linear superposition over the code words of the *dual* code $\mathcal{C}^\perp$, where $\mathcal{C}^\perp=\{u:\ u\cdot v=0\ \forall v\in\mathcal{C}\}$. This can derived when looking at the action of the Hadamard transform on n-bit strings $|x\rangle$:

$$H^{\otimes n}|x\rangle = \frac{1}{\sqrt{2^n}}\sum_{y\in\{0,1\}^n}(-1)^{x\cdot y}|y\rangle$$

As it is easy to see from Eq. (8), the Hamming code is its own dual, and hence we can use its properties to correct phase errors. In general, the CSS constructions find a code $\mathcal{C}_1$ (for the bit flip errors) such that its dual, $\mathcal{C}_1^\perp$, contains a sufficiently good code $\mathcal{C}_2$ (for the phase flip errors).

Having seen the nine qubit Shor code and the seven qubit Steane code, one can ask what the minimal overhead for a quantum code that corrects a single error is. It turns out that the smallest quantum code that achieves this has five qubits, and that this is optimal [LMPZ96].

Gottesman developed a very powerful formalism, so called *stabilizer codes*, that generalizes both the Shor code and CSS codes and gave fault tolerant constructions for them (for more details see App. D).

7 Fault-tolerant computation

"When you have faults, do not fear to abandon them."
Confucius

We have seen that good quantum error correction codes exist. But so far we have worked under the assumption, that the error recovery procedure is *perfect*. Of course, error recovery will never be flawless. Recovery is itself a quantum computation that will be prone to decoherence. We must ensure that errors do not *propagate* during recovery. For instance, if an error occurs in the ancilla bit in the parity measurement of Fig. 5, *all* six qubits interacting with it might be corrupted; the error propagates catastrophically. In fault-tolerant computing design, care is taken to avoid this type of error spreading, and other possible introduction and propagation of error. But even if we manage to avoid error spreading during recovery, that is not enough. A quantum computer does more than just *store* information, it also *processes* it. Of course we could decode, perform a gate and encode, but this procedure would temporarily expose quantum information to decoherence. Instead, we must apply our quantum gates directly to the encoded data.

7.1 Guidelines of fault-tolerance

The quantum circuit model gives us a good intuition about the points in a computation that potentially can introduce errors and corrupt the computation. We need to be able to faultlessly *prepare* the initial state, *compute* with a sequence of quantum gates and *measure* the output. Using a code to protect our computation against noise, we also need to assure faultless *encoding*, *decoding* and *correction*. Each qubit will be encoded into a separate block and quantum logic has to be applied directly *on the encoded states* so that the information is never exposed to noise without protection. This gives us the following guidelines of fault-tolerance:

Encoding/Decoding/State Preparation The procedure to encode/decode the information into a code should not introduce more errors than the code can correct. In the case of a 1-error correcting QECC encoding should not introduce more than one error per encoded block. Often the only states that need to be encoded are some $|00\dots0\rangle$ states at the beginning of the computation, it is then sufficient to ensure fault-tolerant *state-preparation*.

Error-detection and Recovery These procedures (for a QECC), usually realized by a set of quantum gates together with auxiliary qubits, should again not introduce more than one error per block.

Quantum gates should not introduce more than one error per encoded block. Furthermore they should not *propagate* already existing errors from one qubit to several others in the same block.

Measurement should not introduce more than one error per block. Furthermore the measurement result must have probability of error of order ε^2, where ε is the probability of failure of any of the components in the measurement procedure. This is because the measurement result may be used to control other operations in a quantum computer.

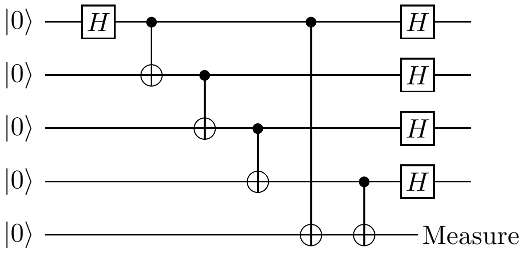

Figure 7: Construction and verification of the $|ancilla\rangle$ state. If the measurement outcome is 1, then the state is discarded and a new $|ancilla\rangle$ state is prepared.

7.2 Fault-tolerant error correction

With these guidelines in place, we will now illustrate how fault-tolerant recovery can be achieved, using the Steane code in Eq.(9) as our example. The error-measurement circuit in Fig. 6 is not fault-tolerant, as each of the CNOT gates can propagate a single phase error on the ancilla qubit to all four of the code qubits. To prevent this propagation, we need to expand the ancilla into four qubits, each one the target of only one CNOT gate. But now we are again faced with the problem that our measurement should only reveal information about the *error* (the parity) but not about the encoded state. We circumvent this problem by preparing the ancilla in the following state

$$|ancilla\rangle = \frac{1}{\sqrt{8}} \left(|0000\rangle + |1100\rangle + |1010\rangle + |1001\rangle + |0110\rangle + |0101\rangle + |0011\rangle + |1111\rangle\right),$$

i.e. in a superposition of all even bit strings. The crucial observation is that on this state one bit flip or three bit flips on any qubits all have the same effect, they transform it to the superposition of odd bit strings. Similarly, this state is invariant under any even number of bit flips. This means that we can infer the syndrome bit from the parity of the ancilla bits, it suffices to measure the ancilla in the end. Hence our syndrome measurement obeys the guidelines of fault-tolerance. To prepare the ancilla, we can use the circuit in Fig. 7, which at the same time allows *verification* of correct ancilla preparation.

The ancilla state must be verified before it is used, because a single error in the preparation of the ancilla state can propagate and cause two phase errors in the $|ancilla\rangle$ state. Hence the circuit in Fig. 7 also verifies that multiple phase errors do not occur. If it fails the test it should be discarded, and the preparation procedure repeated.

Moreover, a single syndrome measurement might be faulty. Thus, the syndrome measurement should be repeated for accuracy; only if the same result is measured twice in a row should it be accepted.

With all the precautions above, recovery will only fail if two independent errors occur in this entire procedure. The probability that this happens is still $c\varepsilon^2$ for some constant c, but because there are now many more gates and steps involved the constant c can be quite large.

In a conceptually similar fashion it is possible to *encode* a qubit and to measure it in a basis spanned by $|0\rangle_{code}$ and $|1\rangle_{code}$ while following the guideline of fault-tolerance. For details the reader should consult e.g. [Sho96, Ste97, Pre98b, Pre99, Pre98a] or the work of Gottesman (e.g. [Got97c]) for fault-tolerant construction for CSS and other codes in the stabilizer formalism (see App. D).

7.3 Fault-tolerant computation

We have seen how to recover *stored* quantum information, even when recovery is faulty. But we also want to *compute*, and the gates we use will be faulty as well. This means that we must be able to apply the gates directly to the encoded data, without introducing errors uncontrollably and following the guidelines of fault-tolerance.

In fact, staying with the 7-qubit Steane code, it is easy to implement some single qubit gates directly on the encoded data. We have seen that the bitwise Hadamard transform implements an *encoded* Hadamard transform on the codewords (see Eq. (10)). This means we can apply it without propagating errors and such that each gate introduces at most one new error. Similarly, it is easy to see that the bitwise X gate induces an encoded X_{enc} because even code words get mapped to odd ones and vice versa. Moreover the bitwise Z gate (which is just HXH) implements the encoded Z. In the same way the $\frac{\pi}{4}$ gate (a diagonal single qubit unitary with diagonal $(1, i)$) can be implemented by applying it bitwise to the encoded data.

Also, it is not hard to see that the bitwise CNOT between two quantum code words, i.e. a CNOT from the first qubit of the first code word to the first qubit of the second code word, a CNOT from the second qubit of the first code word, to the second qubit of the second and so on, implements a global CNOT between two code words. We call such an implementation of an encoded two qubit gate *transversal.* This is very promising, but the set of operation we can implement fault-tolerantly is not yet universal. We also need to implement the $\pi/8$ gate for a universal set of gates. Unfortunately it seems to be impossible to implement the $\pi/8$ gate in a fault-tolerant way. There are several ways to circumvent this problem. Shor, for instance, gave a way to complete the universal set by giving a transversal implementation of a three qubit gate, the Toffoli gate [Sho96]. However, we will follow a slightly different route here. It turns out that the gates $\{X, \frac{\pi}{4}, CNOT\}$ are universal, provided we can measure a code word in the $|0\rangle_{code}$, $|1\rangle_{code}$ basis and we have access to the state

$$|\pi/8_{+}\rangle_{code} = |0\rangle_{code} + \exp(i\frac{\pi}{4})|1\rangle_{code}.$$

It has been shown that there is a fault-tolerant preparation and verification procedure for the state $|\pi/8_{+}\rangle_{code}$, which is similar in spirit to the one in Fig. 7.

Several variants of fault-tolerant universal quantum computation have been developed for this and other codes, like CSS codes and stabilizer codes. They differ in the details of ancilla preparation and number of interactions with the code word. As a result it is possible to implement computation and error correction following the guidelines of fault-tolerance.

8 Concatenated coding and the threshold

"Much of modern art is devoted to lowering the threshold of what is terrible."
Susan Sontag

We have seen how to encode quantum data, how to perform fault-tolerant recovery and how to compute fault tolerantly on encoded states. However, this is still not sufficient to implement quantum algorithms. Quantum codes exist that can correct up to t errors, where t can be as large as we wish, and on which we can compute fault-tolerantly. This means that if our error rate and gate and measurement failure rate is ε, then computation will only fail with probability of order ε^{t+1} for a t of our choice. So what is the problem?

The crux is the complexity of the recovery procedure. With large t we reach a point where the recovery procedure takes so much time that it becomes likely that $t + 1$ errors occur in a block. The number of steps required for recovery scales as a power of t, t^a with exponent $a > 1$. That means that the probability to have $t + 1$ errors before a recovery step is completed, scales as $(t^a\varepsilon)^{t+1}$. This expression is minimized when $t = c\varepsilon^{-\frac{1}{a}}$ for some constant c and its value is at least $p_{fail} = \exp(-ca\varepsilon^{-\frac{1}{a}})$. This means that per error correction cycle our probability to fail is at least p_{fail}. If we have N such cycles, our failure probability is $Np_{fail} = \exp(-ca \log N\varepsilon^{-\frac{1}{a}})$. If we want to keep this (much) smaller than 1, our error rate ε has to scale as $\frac{1}{(\log N)^a}$, i.e. the longer the computation, the more accuracy we need; an unrealistic assumption.

To overcome this problem, a special kind of hierarchical approach is used [KLZ98] (see also [Kit97, A]). Ideas related to this approach go back to pioneering works of John von Neumann, who established a theory of fault-tolerant computation for noisy classical computers [Neu56].

Suppose that we encode our information into a code, like Steane's code. Then, in turn, we encode each qubit of this encoded qubit using again Steane's code, and so on. We obtain several layers of encoded qubits, say k layers, and the total number of qubits is 7^k. This type of code is called *concatenated* code.

The exact calculations behind the threshold theorem are rather intricate. Let us only give a rough intuition. The idea is to perform error recovery most often at the lowest level, and less and less often at higher levels of the hierarchy, which have more qubits. We recursively apply the idea of simulating a circuit using an encoded circuit, constructing a hierarchy of quantum circuits. Suppose in the first stage the original qubit is encoded in a quantum code whose encoded qubits are again encoded in a quantum code and so on. Each level has some error recovery cycles. If the failure probability at the lowest level of this code is ε then the failure probability at the next level of encoding is $c\varepsilon^2$ (remember that the Steane code reduces the error rate from first to second order), where c counts all possibilities that two errors can occur, given the number of gates in the recovery procedure and the fault-tolerant application of gates. Continuing with this reasoning, the *effective* error rate at the next level is $c\varepsilon^2$, and error recovery reduces the error to $c(c\varepsilon^2)^2$. Proceeding level by level, we see that at the kth level of the hierarchy an error on one of the sub-blocks only has probability $(c\varepsilon)^{2^k}/c$. We see that if our noise rate is below a certain threshold, $\varepsilon < \varepsilon_{th} \equiv 1/c$, then the error is reduced in each level of concatenation. This gives the *error threshold* for fault-tolerant quantum computation.

How does the total size of the circuit grow? Let's assume that one level of encoding requires an overhead of G gates to fault-tolerantly perform a gate and error-correct. Then the size of the simulating circuit grows as G^k. Let us see when this concatenation procedure gives a small enough failure probability:

Assume the initial quantum circuit we want to emulate has N gates and we wish to achieve a final accuracy $p_{success}$. In such a circuit each gate has to be accurate to $p_{success}/N$ (gate errors add linearly). To achieve this we concatenate k times so that

$$\frac{(c\varepsilon)^{2^k}}{c} = \varepsilon_{th}(\frac{\varepsilon}{\varepsilon_{th}})^{2^k} \leq \frac{p_{success}}{N}$$

or

$$2^k \leq \frac{\log(N\varepsilon_{th}/p_{success})}{\log(\varepsilon_{th}/\varepsilon)}$$

If ε is smaller than the threshold value, such a k can be found. For error rates below the threshold we can achieve arbitrary accuracy by concatenation. Per initial gate the final circuit will have

$$G^k = 2^{k\log G} \leq \left(\frac{\log(N\varepsilon_{th}/p_{success})}{\log(\varepsilon_{th}/\varepsilon)}\right)^{\log G} = poly(\log N)$$

gates and so its final size will be $Npoly(\log N)$ which is only polylogarithmically larger than the original N.

Note that we have crudely simplified our calculations. Estimating the threshold is an extremely intricate task. Its value depends on the details of the code and fault-tolerance constructions used. It also depends on whether we assume the classical syndrome processing to be perfect or not. In all cases it seems that we need high parallelization and a supply of fresh ancilla qubits during the computation. For a long time the actual value of the threshold has been estimated by optimists and pessimists to lie somewhere between 10^{-4} and 10^{-7}. Recent work seems to indicate that it can be even as high as 3% [Kni05] (see also [AGP05, Rei05]) and optimized numerical simulations of fault-tolerant protocols suggest a threshold as high as 5% (however, to tolerate this much error existing protocols require enormous overhead).

9 Error avoidance and Decoherence Free Subsystems

"It is well known that "problem avoidance" is an important part of problem solving."
Edward de Bono

In all our previous analysis we have assumed that the errors behave independently and affect few qubits at a time. What, if this is not the case? There are situations, where groups of qubits interact with the environment in a *collective* fashion, possibly undergoing a correlated error. For these cases the theory of decoherence-free subspaces and subsystems has been developed, sometimes also called *error-avoiding* codes. These codes come into play when the decoherence process is in some sense not local, but collective, involving groups of qubits.

Let us give a classical example. Assume we have an error process that with some probability flips all bits in a group, and otherwise does nothing. In this case we can encode a classical bit as

$$0 \longrightarrow 00 \qquad 1 \longrightarrow 01.$$

The error process will change the encoded states to

$$00 \longrightarrow 11 \qquad 01 \longrightarrow 10.$$

But no matter if the error has acted or not, the *parity* of the bit string is unchanged. So when we decode, we will associate 00 and 11 with the encoded 0 bit and 01 and 10 with the encoded 1. Note that we will be able to decode correctly *no matter* how hight the rate of error is! The error does not touch the invariant, parity, into which we encode. That means that our encoded information has managed to completely *avoid* the error, we have given the simplest error-avoiding code.

A lot of research has been done to generalize this to the quantum case (see e.g. [Kem01, Bac01, LW03] for surveys). The noise model is in general derived from the Hamiltonian picture (see App. B.1) or from the Markovian picture (see App. B.3), a brief derivation is given in App. E. In general the underlying assumption is that several qubits couple collectively to the environment and are affected by a symmetric decoherence process. In systems where this form of decoherence is dominant at the qubit level, error-avoiding codes as part of the error-correction scheme are advantageous.

We will content ourselves with briefly describing one example. For one of the most common collective decoherence processes the noise operators on n qubits (see App. E) in the Hamiltonian picture (App. B.1) are given by $S_\alpha = \sum_{i=1}^{n} \sigma_\alpha^i$, where σ_α^i is a Pauli matrix ($\alpha = \{x, y, z\}$) on the ith qubit. Intuitively this means that the possible unitary errors are $\exp(itS_\alpha)$. The condition for decoherence-free subspaces is that

$$S_\alpha |codeword\rangle = c_\alpha |codeword\rangle,$$

or in other words that the code space is a simultaneous eigenspace of each S_α with eigenvalue c_α. If this is the case, each unitary noise operator only introduces an unobservable phase $exp(itc_\alpha)$ on the code space.

Let us look at an encoding of 4 qubits:

$$\begin{aligned} |0\rangle_{code} &= |s\rangle \otimes |s\rangle \\ |1\rangle_{code} &= \frac{1}{\sqrt{3}} \left(|t_+\rangle \otimes |t_-\rangle - |t_0\rangle \otimes |t_0\rangle + |t_-\rangle \otimes |t_+\rangle \right), \end{aligned}$$

where $|s\rangle = \frac{|01\rangle - |10\rangle}{\sqrt{2}}$ and $|t_{-,0,+}\rangle = \{|00\rangle, \frac{|01\rangle + |10\rangle}{\sqrt{2}}, |11\rangle\}$. It is easy to see that $S_\alpha|0\rangle_{code} = S_\alpha|1\rangle_{code} = 0$ for $\alpha = \{x, y, z\}$ (i.e. that the coefficients $c_\alpha = 0$). This means that both code states are invariant under collective noise. If we encode our information into the subspace spanned by $|0\rangle_{code}$ and $|1\rangle_{code}$, it will completely avoid the errors; it resides in a “quiet” part of the space, a *decoherence-free* subspace.

This idea has been generalized to a wide variety of encodings (in particular to decoherence-free *subsystems*, see App. E for a little more detail) and against several kinds of collective noise. It has been shown how to *compute* on these codes (e.g. [KBLW01]) and how to use them in a fault-tolerant framework.

Of course the noise in a real implementation of a quantum computer will be a mixture of independent statistical noise, and coupled collective noise, depending on the specific quantum hardware used. The idea is to use a hierarchical construction of concatenated quantum codes, as in the threshold construction of Sec. 8, where the lower levels of the hierarchy use error correction (or avoidance) schemes that are highly specialized to the anticipated noise process, whereas higher levels are similar to the known fault-tolerant constructions for QECCs (see e.g. [LBW99] for a DFS-QECC concatenation scheme).

10 Conclusion and Epilogue

"The best thing about the future is that it only comes one day at a time."
Abraham Lincoln

Without doubt work on quantum fault-tolerance is of prime importance if we want to build quantum machines. In the late 1990's pioneering work has established that fault-tolerant quantum computation is *possible*, and we have estimates for the *error threshold*, the maximum error a component can undergo such that the computation still proceeds without catastrophic error. On the way we have gained new insights into the nature of decoherence and about the methods and tools used to model and describe it.

At this point we need to optimize the details of fault-tolerant schemes and to generate new ideas to improve the threshold. Current experimental results show that the accuracy in implementations of the quantum circuit model is on the order of several percent in the best case, whereas most estimates of the threshold give numbers of the order of 10^{-4} or less.

The task for current research is to analyze the threshold for particular codes and to develop new elements of fault-tolerance that improve the threshold. For instance only recently the existence of a threshold for the Steane code has been shown [AGP05, Rei05]. New elements have been developed to improve the threshold, like for instance schemes based on postselection [Kni04]. Using new ingredients, the threshold has now been estimated to be on the order of 3%, albeit with an enormous overhead in the circuit architecture [Kni05].

Another avenue of research is to explore other models of quantum computing, different from the quantum circuit model, which can be inherently more robust against noise. One idea, initiated by Kitaev [Kit03], is a scheme for intrinsically fault-tolerant quantum hardware, designed to be robust against localized inaccuracies. In this scheme (quantum computing by anyons) gates exploit non-Abelian Aharonov-Bohm interactions among separated quasiparticles on a 2D lattice (see also e.g. [SBFH05]).

Another potentially more robust model is the model of adiabatic quantum computation, where computation is achieved by adiabatically tuning a set of Hamiltonians, and where the system is always in the instantaneous groundstate. In this system there is a gap between the groundstate and the first exited state at all times, which might make the state more robust to noise (see [FGG$^+$01, AvDK$^+$04, CFP02]). Yet another model is the measurement based *one-way* quantum computer [BR01]. Here quantum computation is achieved by measuring single qubits of a suitably prepared initial state. The fault-tolerance properties of this system have recently been explored in [ND05].

All these developments allow us to be optimistic about the future of a quantum computing machine. One day we might be able to combat decoherence and have large scale entangled states operate for us. The consequences not only for computation, but also for our understanding of the fundamental processes behind decoherence will be formidable.

A Further Reading

An ever growing community of researchers has been and is working an error correction and fault-tolerance in the quantum setting, and it is impossible to mention all of them in this framework. What follows is a selection of some of the milestones and recent developments, where the interested reader can find more information.

That quantum error correcting codes exist was first pointed out by Shor [Sho95] and Steane [Ste96b] in the end of 1995. By early 1996 it was shown by Steane [Ste96a] and Calderbank and Shor [CS96] that *good* codes exist, i.e. codes that are capable to correct many errors. The quantum error correction conditions where formalized by Knill and Laflamme [KL97] and Bennett et al. [BDSW96][4].

The first fully *fault-tolerant recovery* scheme, which takes into account that encoding, error-correction and decoding are themselves noisy operations, was developed by Shor in 1996 [Sho96].

[4] These authors also analyzed schemes based on random codes.

Methods for fault-tolerant recovery where independently developed by Kitaev.

The first to show that there is an *accuracy threshold* for *storage* of quantum information where Knill and Laflamme [KL96] and for *quantum computation* Knill, Laflamme and Zurek [KLZ96] in 1996 (see also [KLZ98]). Similar results were reported by Kitaev [Kit97] and Aharonov and Ben-Or [A].

The theory of *stabilizer codes* and of fault-tolerance in the powerful stabilizer formalism was developed by Gottesman (see e.g. [Got97c])[5].

Since then several researchers have sharpened the threshold and developed new techniques to analyze and improve it. See for instance the recent work of Steane [Ste99, Ste03], Knill [Kni04, Kni05], Aliferis, Gottesman and Preskill [AGP05] and Reichardt [Rei05] and [FKSS04] for a dynamical systems approach.

In the literature the terms "sub-" and "superradiance" are often encountered in connection with *collective decoherence* processes. Decoherence-free subspaces have been studied by several researchers (see for example [DG98, ZR97, Zan97, Zan98] in the context of storage, and [LCW98, BLW99, BKLW00, KBLW01, DBK$^+$00] in the context of fault-tolerant computation and [LBW99] in combination with QECCs). Decoherence-free subsystems have been introduced by [VKL99, KLV00], also in the connection with dynamic decoupling techniques. Since then there has been a lot of active effort to adapt codes to various collective noise processes.

The threshold has also been inspected in the light of various (local) error models. For instance [TB05] discuss the fault-tolerant threshold for local *non-Markovian* noise (see also [ALZ05] and references therein for a recent controversy about the nature of errors and an analysis of error models that seem to not allow fault-tolerant computation in [Kal05]).

Apart from the quantum circuit model alternative proposals for quantum architectures have been developed, which could be potentially more robust than the quantum circuit model. One example, developed by Kitaev, is the model of computation via *anyons* (see [Kit03] for an analysis of its fault-tolerance properties, or e.g. Freedman et al. [FKLW01]). Another recent example is the measurement based *quantum cluster* model [BR01] introduced by Briegel and Raussendorf. Recently, fault-tolerance has been analyzed in this model by Nielsen and Dawson [ND05]. The *adiabatic model* of quantum computation has been introduced by Farhi et al. [FGG$^+$01] and shown to be equivalent to the quantum circuit model in [AvDK$^+$04]. Childs et al. have discussed its robustness in [CFP02].

B How to model decoherence

No quantum system can be perfectly isolated from its surroundings and be viewed as perfectly closed. In the physical world, degrees of freedom are usually interacting with many other degrees of freedom. In fact, the understanding of this point is crucial for the explanation of why classical mechanics in the macroscopic world emerges out of the microscopic operation of quantum mechanics. Even if we find quantum computer elements that interact only weakly with the rest of the world (achievable most likely if they are themselves of atomic or near-atomic dimensions), for short times the evolution will be unitary, but eventually even weak interactions will cause significant departure from unitarity. Physical systems have a characteristic time for loss of unitarity, which is known in the field of mesoscopic physics as the "dephasing time". It is often extremely short (for a table of dephasing times for various systems see [DiV95]), for example for the state of an electron traversing a gold wire at temperature less than 1K it is of order 10^{-13} seconds.

We refer to the effects of noise due to unwanted coupling with the environment as decoherence[6]. An early treatise on quantum noise from a rather mathematical point of view is due to

[5] See also work by Calderbank, Rains, Shor and Sloane, which develop stabilizer codes as codes over GF(4) [CRSS98]

[6] An unfortunate confusion in terms has arisen with the word "decoherence". Historically it has been used to refer just to a phase damping process - a specific type of noise - cf. e.g. Zurek [Zur91]. Zurek and others realized the unique role played by phase damping in the transition from a quantum to a classical world. However, in the quantum computing community by and large the term "decoherence" is now used to refer to *any noise process* in quantum processing.

Davies [Dav76]. Caldeira and Leggett [CL83] in 1983 undertook one of the first and most complete studies of an important model, the *spin-boson model*.

Within the context of quantum computers these studies were taken up by Unruh [Unr95] in 1995 and developed by many others (e.g. Palma et al. [PSE96], Zanardi [Zan97, Zan98]). Over the past few years work on quantum computation has generated profound insights into the nature of decoherence.

B.1 Hamiltonian Picture

To model the dynamics of a register of qubits (quantum computer) with its surroundings we imagine the system immersed into its environment (often called bath) and the whole (quantum register plus environment) as a closed system described in a general way by the following Hamiltonian:

$$\mathbf{H} = \mathbf{H}_S \otimes \mathbf{I}_B + \mathbf{I}_S \otimes \mathbf{H}_B + \mathbf{H}_I, \tag{11}$$

where $\mathbf{H}_S$ ($\mathbf{H}_B$) [the system (bath) Hamiltonian] acts on the system (bath) Hilbert space $\mathcal{H}_S$ ($\mathcal{H}_B$), $\mathbf{I}_S$ ($\mathbf{I}_B$) is the identity operator on the system (bath) Hilbert space, and $\mathbf{H}_I$, which acts on both the system and bath Hilbert spaces $\mathcal{H}_S \otimes \mathcal{H}_B$, is the interaction Hamiltonian containing all the nontrivial couplings between system and bath. In general $\mathbf{H}_I$ can be written as a sum of operators which act separately on the system ($\mathbf{S}_\alpha$'s) and on the bath ($\mathbf{B}_\alpha$'s):

$$\mathbf{H}_I = \sum_\alpha \mathbf{S}_\alpha \otimes \mathbf{B}_\alpha. \tag{12}$$

(Note that this decomposition is not necessarily unique.)

In the absence of an interaction Hamiltonian ($\mathbf{H}_I = 0$), the evolution of the system and the bath are separately unitary: $\mathbf{U}(t) = \exp[-i\mathbf{H}t] = \exp[-i\mathbf{H}_S t] \otimes \exp[-i\mathbf{H}_B t]$ (we set $\hbar = 1$ throughout). Information that has been encoded (mapped) into states of the system Hilbert space remains encoded in the system Hilbert space if $\mathbf{H}_I = 0$. However in the case when the interaction Hamiltonian contains nontrivial couplings between the system and the bath, information that has been encoded over the system Hilbert space does not remain encoded over solely the system Hilbert space but spreads out instead into the combined system and bath Hilbert space as the time evolution proceeds.

Very often to describe decoherence in more specific contexts [Unr95, PSE96] it is convenient to model the environment (the bath) as a mass-less scalar field, usually assumed to be in a thermal state (described by a density matrix in Fock-space, the state space used to model *fields*. Its (infinite dimensional) Hilbert space is spanned by

$$\bigotimes_{\mathbf{k}} |i\rangle_{\mathbf{k}} \quad i \in \{0, 1, 2, \ldots\}$$

where $\mathbf{k}$ labels the *modes*. On each mode (factor in the tensor product) we have two operators, the *lowering operator* $\mathbf{b}_{\mathbf{k}}$ given by $\mathbf{b}_{\mathbf{k}}|i\rangle_{\mathbf{k}} = \sqrt{i}|i-1\rangle_{\mathbf{k}}$ and the *raising operator* $\mathbf{b}_{\mathbf{k}}^\dagger$ given by $\mathbf{b}_{\mathbf{k}}^\dagger|i\rangle_{\mathbf{k}} = \sqrt{i+1}|i+1\rangle_{\mathbf{k}}$. Note that $\mathbf{b}_{\mathbf{k}}^\dagger\mathbf{b}_{\mathbf{k}}|i\rangle_{\mathbf{k}} = i|i\rangle_{\mathbf{k}}$, i.e. $|i\rangle_{\mathbf{k}}$ is an *eigenstate* of the *number operator* $\mathbf{b}_{\mathbf{k}}^\dagger\mathbf{b}_{\mathbf{k}}$.

The quantum register is described by an arrangement of n two-level systems (spins). This results in the *spin-boson model*, where the bath Hamiltonian can be written as

$$\mathbf{H}_B = \sum_k \omega_k \mathbf{B}_k$$

and, e.g., for the spin-boson Hamiltonian, $\mathbf{B}_k = \mathbf{b}_k^\dagger \mathbf{b}_k$ [LCD$^+$87], and $\mathbf{b}_k^\dagger$, $\mathbf{b}_k$ are respectively creation and annihilation operators of bath mode k. The interaction Hamiltonian is given by

$$\mathbf{H}_I = \sum_{i=1}^{n} \sum_{\alpha=+,-,z} \sum_k g_{ik}^\alpha \sigma_\alpha^i \otimes \tilde{\mathbf{B}}_k^\alpha + h.c., \tag{13}$$

where g^{α}_{ik} is a coupling coefficient and *h.c.* denotes the hermitian conjugate. In the spin-boson model one would have $\tilde{\mathbf{B}}_k^+ = \mathbf{b}_k$, $\tilde{\mathbf{B}}_k^- = \mathbf{b}_k^\dagger$ and $\tilde{\mathbf{B}}_k^z = \mathbf{b}_k^\dagger + \mathbf{b}_k$. Thus $\sigma_\pm^i \otimes \tilde{\mathbf{B}}_k^\pm$ expresses a dissipative coupling (in which energy is exchanged between system and environment), and $\sigma_z^i \otimes \tilde{\mathbf{B}}_k^z$ corresponds to a phase damping process (in which the environment randomizes the system phases, e.g., through elastic collisions).

B.2 Operator Sum Picture

The evolution of a quantum state in the entire space is unitary in a closed system of which we can observe and control all parts. Very often, however, this is not the case: imagine for example that we perform a certain measurement and then "forget" or lose the measurement outcome. As a result we know that the state has collapsed into some eigenstate of of the measurement operator, but not into which one, and we will have to assign probabilities to each of them to model the current state of the system. Take the example of a qubit $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ and the measurement in the computational basis. Performing this measurement and "throwing away" the result will leave the system in the state $|0\rangle$ with probability $|\alpha|^2$ and in the state $|1\rangle$ with probability $|\beta|^2$. To describe this *mixture* of possible states the *density matrix* formalism proved to be very useful: we write

$$\rho = |\alpha|^2|0\rangle\langle 0| + |\beta|^2|1\rangle\langle 1| = \begin{pmatrix} |\alpha|^2 & 0 \\ 0 & |\beta|^2 \end{pmatrix}$$

Another way we can think about density matrices is to imagine that we have a (big) quantum system and can access only part of it. To describe the quantum state of the accessible part (call it A), we have to average over the non- accessible degrees of freedom of the system (part B). This is done by performing a complete measurement on system B (mentally) and "throwing away" the outcomes (because we do not have access to them). Let us give the example of a state of 2 qubits $|\psi\rangle_{AB} = \alpha|00\rangle + \beta|11\rangle$ where we only have access to the first qubit (part A). If we (mentally) measure system B in the computational basis, we obtain the density matrix from Eq. (14).

Let us proceed to the more general description of the statics and dynamics of open quantum systems, described by mixed states:

States: States in an N-dimensional Hilbert space $\mathcal{H}_N$ are given by density matrices ρ such that:

- ρ is hermitian: $\rho^\dagger = \rho$
- ρ is positive: $\forall|\psi\rangle \in \mathcal{H}_N \quad \langle\psi|\rho|\psi\rangle \geq 0$, which is equivalent to $\lambda_i \geq 0$, where λ_i are the eigenvalues of ρ (this can be viewed as a statement about the positivity of probabilities of the pure states in the mixture).
- ρ has trace 1 (this corresponds to the normalization of probabilities)

Pure states $|\psi\rangle$ of the system are associated with the density matrix $\rho_{pure} = |\psi\rangle\langle\psi|$. A general mixed state is diagonalizable and can be written in its spectral decomposition as

$$\rho = \sum_k p_k |\psi_k\rangle\langle\psi_k|$$

Note that there are in general many other ways to write ρ in the above form if we allow for non-orthogonal states in the decomposition. Each such decomposition $\{q_k, |\phi_k\rangle\}$ is called an *ensemble realization* of ρ. The ambiguity in the decomposition of ρ manifests some loss of information, in the sense that the probabilistic mixture ρ could have arisen in a multitude of ways.

Dynamics: To describe the evolution of an open system - and thus of decoherence - we will immerse it into a closed system. The evolution of a closed system is described by a unitary transformation, which translates to an effective dynamics of the open system governed by completely positive (and trace-preserving) maps. Loosely speaking, these are maps that in $\mathcal{H}_A$ take density matrices to density matrices, with the additional property that if we extend the map to bigger spaces $\mathcal{H}_{AB}$ by applying the map on A and the identity map on B, then they still take density matrices to density matrices. More precisely, an operator map Λ is completely positive if $(I \otimes \Lambda)(\rho) \geq 0$ whenever $\rho \geq 0$.

The important point here is that according to *Kraus' Representation Theorem* [Kra83] every completely positive trace preserving map can be written as

$$\rho \to \sum_{\mu} M_{\mu} \rho M_{\mu}^{\dagger} \quad \text{with} \ \sum_{\mu} M_{\mu}^{\dagger} M_{\mu} = I \tag{14}$$

where the M_{μ} are N-by-N matrices (N being the dimension of the Hilbert space). In particular this describes both the Hamiltonian and the Markovian dynamics, though in general it is often tedious to derive the form of the M_{μ} from the $\mathbf{S}_{\alpha}$ of the Hamiltonian picture (Eq. (12)). Note that contrary to the case of unitary evolution, general open system dynamics is not *reversible.*

B.3 Markovian Picture

Another very powerful formalism to describe decoherence is the approach of *master equations. Markovian quantum dynamics* describes processes resulting from the interaction with a Markovian environment in the so called Born-Markov approximation. The main objective is to describe the time-evolution of an open system with a differential equation - the *Master equation* - which properly describes non-unitary behavior.

In fact it is not a priori obvious that there needs to be a differential equation that describes decoherence. Such a description will be possible only if the evolution of the quantum system will be local in time (*Markovian*), i.e. that the density operator $\rho(t + dt)$ is completely determined by $\rho(t)$. This is usually not the case because the bath retains a memory of the state of ρ at previous times for a while and can transfer it back to the system.

To obtain the Master equation in the Born-Markov approximation a common approach is to start with the Hamiltonian description Eq. (11) and use time-dependent perturbation theory (i.e. an expansion into time-series) with careful truncation (cf. [Car93]).

A more axiomatic way, followed by Lindblad [Lin76, AL87], is to establish the most general linear equation for density matrices. More precisely, by assuming that (i) the evolution of the system density matrix is a one-parameter semigroup, (ii) the system density matrix retains the properties of a density matrix including "complete positivity", and (iii) the system and bath density matrices are initially decoupled, Lindblad [Lin76] has shown that the most general evolution of the system density matrix $\rho_S(t)$ (in a Hilbert space of dimension N) is governed by the master equation

$$\begin{aligned} \frac{d\rho}{dt} = \mathbf{L}[\rho] &= -i[\mathbf{H}_S, \rho] + \frac{1}{2} \sum_{\alpha,\beta=1}^{M} a_{\alpha\beta} \left([\mathbf{F}_{\alpha}, \rho \mathbf{F}_{\beta}^{\dagger}] + [\mathbf{F}_{\alpha}\rho, \mathbf{F}_{\beta}^{\dagger}] \right) \\ &= -i[\mathbf{H}_S, \rho] + \frac{1}{2} \sum_{\alpha,\beta=1}^{M} a_{\alpha\beta} \mathbf{L}_{\alpha,\beta}[\rho]. \end{aligned} \tag{15}$$

Here $\mathbf{H}_S$ is the system Hamiltonian generating unitary evolution plus possible additional terms due to the interaction with the bath - usually referred to as *Lamb-shift* -; the operators $\mathbf{F}_{\alpha}$ constitute a basis for the M-dimensional space of all bounded operators acting on $\mathcal{H}_S$[7], and $a_{\alpha\beta}$ are the elements of a positive semi-definite Hermitian matrix. We refer to the matrix $a_{\alpha\beta}$ as the *GKS matrix.*

Every such process described by Eq. (15) corresponds to some interaction which, if applied for a duration t, induces a quantum operation $\mathcal{E}_t$. The class of quantum operations $\mathcal{E}_t$ forms a Markovian semigroup, such that

$$\mathcal{E}_s \mathcal{E}_t = \mathcal{E}_{s+t} \, .$$

[7] they are often called the *Lindblad operators* or the *quantum jump operators*

Here $\mathcal{E}_s\mathcal{E}_t$ denotes composition of the operations, i.e., $\mathcal{E}_s \circ \mathcal{E}_t$. Each Markovian semigroup in turn describes the dynamics resulting from some interaction with a Markovian environment in the Born approximation.

Note that the Operator Sum Representation also describes Markovian dynamics, though it is in practice often difficult to derive the M_μ (Eq. (14)) from the $\mathbf{F}_\alpha$ of the Markovian picture (Eq. (15)).

To make our description of Markovian quantum dynamics concrete, we present some important examples of qubit noise processes[8]. We choose the basis $\{F_\alpha\}$ to be the normalized Pauli operators $\frac{1}{\sqrt{2}}\{\sigma_x, \sigma_y, \sigma_z\}$, and we write the density matrix of a qubit as

$$\rho = \begin{pmatrix} \rho_{00} & \rho_{01} \\ \rho_{10} & \rho_{11} \end{pmatrix}.$$

The first process, *phase damping*, acts on a qubit as

$$\mathcal{E}_t^{\mathrm{PD}}(\rho) = \begin{pmatrix} \rho_{00} & e^{-\gamma t}\rho_{01} \\ e^{-\gamma t}\rho_{10} & \rho_{11} \end{pmatrix},$$

where γ is a decay constant and t is the duration of the process. The generator has a GKS matrix with $a_{33}^{\mathrm{PD}} = \frac{\gamma}{2}$ and all other $a_{\alpha\beta}^{\mathrm{PD}} = 0$. The second example is the *depolarizing channel*, which acts on a qubit as

$$\mathcal{E}_t^{\mathrm{DEP}}(\rho) = \begin{pmatrix} \frac{1+e^{-\tilde{\gamma}t}(\rho_{00}-\rho_{11})}{2} & e^{-\tilde{\gamma}t}\rho_{01} \\ e^{-\tilde{\gamma}t}\rho_{10} & \frac{1+e^{-\tilde{\gamma}t}(\rho_{11}-\rho_{00})}{2} \end{pmatrix}.$$

Its GKS matrix has the nonzero elements $a_{11}^{\mathrm{DEP}} = a_{22}^{\mathrm{DEP}} = a_{33}^{\mathrm{DEP}} = \tilde{\gamma}/4$. Our final example is *amplitude damping*, which acts on a qubit as

$$\mathcal{E}_t^{\mathrm{AD}}(\rho) = \begin{pmatrix} \rho_{00} + (1-e^{-\Gamma t})\rho_{11} & e^{-\Gamma t/2}\rho_{01} \\ e^{-\Gamma t/2}\rho_{10} & e^{-\Gamma t}\rho_{11} \end{pmatrix}.$$

The GKS matrix $a_{\alpha\beta}^{\mathrm{AD}}$ is given by

$$\frac{\Gamma}{4}\begin{pmatrix} 1 & -i & 0 \\ i & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}. \tag{16}$$

C The Error-model

The underlying key assumption for efficient usage of quantum error-correcting codes is the *independent error model.* Intuitively, if a noise process acts independently on the different qubits in the code, then provided the noise is sufficiently weak, error-correction should improve the storage fidelity of the encoded over the unencoded state.

Mathematically, the assumption of independent errors can be retraced in each of the decoherence pictures introduced in Sec. B. In the Hamiltonian picture we can rewrite Eq. (13) as

$$\mathbf{H}_I = \sum_{i=1}^{K} \sum_{\alpha=x,y,z} \sum_k \sigma_i^\alpha \otimes \mathbf{B}_{ik}^\alpha,$$

where $\mathbf{B}_{ik}^z \equiv \tilde{\mathbf{B}}_k^z$ and $\mathbf{B}_{ik}^x$,$\mathbf{B}_{ik}^y$ are appropriate linear combinations of $\tilde{\mathbf{B}}_k^+$ and $\tilde{\mathbf{B}}_k^-$:

$$\begin{aligned} \mathbf{B}_{ik}^x &= \frac{1}{2}\left(g_{ik}^-\tilde{\mathbf{B}}_k^- + g_{ik}^+\tilde{\mathbf{B}}_k^+\right) \\ \mathbf{B}_{ik}^y &= \frac{i}{2}\left(g_{ik}^-\tilde{\mathbf{B}}_k^- - g_{ik}^+\tilde{\mathbf{B}}_k^+\right) \end{aligned}$$

[8] For a review of these processes and their relevance to quantum information theory, see [NC00]; [Pre98a].

i.e. all system components can be expressed in terms of tensor products of the single qubit *Pauli matrices.* If we expand the evolution to first order in time and assume that the error-rates g_{ik} are independent we will get an operator sum representation (OSR) (cf. Eq. (14)) where each term is a linear combination of the Pauli matrices (see [LBW99] for a recent derivation). In the Markovian formulation of noise (cf. Eq. (15)) the independent error model assumes that each of the $\mathbf{F}_\alpha$ affects only one of the qubits and that the $\mathbf{F}_\alpha$ are not correlated. Higher order correlations are taken into account by using a code that is suitably constructed for the particular error-model. Therefore the theory of QECCs has focused on searching for codes that make quantum information robust against 1, 2,... or more erroneous qubits, as this is the most reasonable model when one assumes spatially separated qubits with their own local environments. *Detection and correction procedures must then be implemented at a rate higher than the intrinsic error rate.*

¿From the linear decomposition of the error operators in the OSR or the master equation it follows that QECC-schemes need to be able to correct only a discrete set of errors, namely those generated by the Pauli-group[9]. Intuitively we can imagine that the error process acting on one qubit puts the quantum state into a superposition of one of the four possible discrete errors $(I_2, \sigma_{x,y,z})$ and the error-detection and correction procedure collapses the state into one of these errors and then corrects as needed. This intuition can be made formal [NC00, KL97]: it is possible to decompose the the operators M_μ that appear in Eq. (14) into a basis of tensor products of the Pauli matrices. It can then be seen, when deriving the OSR representation from the Hamiltonian picture, that to first order in the noise rate the noise process gives terms with a single non-identity matrix (single qubit error). The core message is again that quantum codes need to account only for a discrete *linear basis* of all possible errors.

D Stabilizer Codes

Stabilizer codes, also known as *additive* codes, are an important subclass of quantum codes. The stabilizer formalism provides an insightful tool to quantum codes and fault-tolerant operations. It was developed by Gottesman in 1996 [Got97a, Got97c]. We will not outline the full formalism here but rather describe only the key elements. A full treatment can be found in [Got97b].

The powerful idea behind the stabilizer formalism is to look at the set of group elements that *stabilize* a certain code and to work with this stabilizer instead of directly with the code. In the framework of QECCs, the stabilizer permits on the one hand to identify the errors the code can detect and correct. It also links quantum codes to the theory of classical error correcting codes in a transparent fashion. On the other hand it also allows one to find a set of universal, fault-tolerant gates.

An operator $\mathbf{S}$ is said to stabilize a code $\mathcal{C}$ if

$$|\Psi\rangle \in \mathcal{C} \quad \text{iff} \quad \mathbf{S}|\Psi\rangle = |\Psi\rangle \quad \forall \mathbf{S} \in \mathcal{S}. \tag{17}$$

The set of operators $\{\mathbf{S}\}$ form a group $\mathcal{S}$, known as the stabilizer of the code [Got97c]. Clearly, $\mathcal{S}$ is closed under multiplication. In the theory of QECC the underlying group is the Pauli-group, the stabilizers are subgroups of the Pauli-group (tensor products of $\mathbf{I}, \mathbf{X}, \mathbf{Y}, \mathbf{Z}$). Since any two elements of the Pauli group either commute or anti-commute, the stabilizer, in this case is always *Abelian.* The code is thus the common eigenspace of the stabilizer elements with eigenvalue 1. *Additive* codes are completely characterized by their stabilizer $\mathcal{S}$. The stabilizer $\mathcal{S}$ can be given by a set of generators which span the stabilizer group via multiplication.

We define the *centralizer* of $\mathcal{S}$ to be the set of elements e in the Pauli group that commute with every element in $\mathcal{S}$, i.e. $e\mathbf{S} = \mathbf{S}e$ for all $\mathbf{S} \in \mathcal{S}$. In case of the Pauli group it coincides with the *normalizer* of $\mathcal{S}$ - the set of elements E in the Pauli-group with $E\mathbf{S}E^\dagger \in \mathcal{S}$ for all $\mathbf{S} \in \mathcal{S}$. We will denote it by $N(\mathcal{S})$ and call it normalizer throughout. Note that the normalizer contains the stabilizer $\mathcal{S}$ itself.

Recall that in the theory of QECCs the error process $\mathcal{E}$ can be expanded in terms of the error basis $\mathbf{E}$ which is a subgroup of the Pauli group. In particular $\mathbf{E}_\alpha \in \mathbf{E}$ either commutes or

[9] In the context of error-correction the Pauli matrices $\sigma_{x,y,z}$ are often denoted by $\mathbf{X}, \mathbf{Y}, \mathbf{Z}$ respectively.

anti-commutes with elements in the stabilizer. This allows us to recast the QECC-condition Eq. (7) in the stabilizer formalism as follows

QECC-conditions: A quantum code $\mathcal{C}$ with stabilizer $\mathcal{S}$ is an $\mathcal{E}$-correcting QECC if for all $\mathbf{E}_\alpha, \mathbf{E}_\beta \in \mathbf{E}$ one of the following holds:

(1) There is an $\mathbf{S} \in \mathcal{S}$ that anti-commutes with $\mathbf{E}_\alpha^\dagger \mathbf{E}_\beta$

(2) $\mathbf{E}_\alpha^\dagger \mathbf{E}_\beta \in \mathcal{S}$.

This clearly implies the QECC conditions Eq. (7), since in the case of (1) $\langle \Psi_i | \mathbf{E}_\alpha^\dagger \mathbf{E}_\beta | \Psi_j \rangle = \langle \Psi_i | \mathbf{E}_\alpha^\dagger \mathbf{E}_\beta \mathbf{S} | \Psi_j \rangle = -\langle \Psi_i | \mathbf{S} \mathbf{E}_\alpha^\dagger \mathbf{E}_\beta | \Psi_j \rangle = 0$ and in the case of (2) $\langle \Psi_i | \mathbf{E}_\alpha^\dagger \mathbf{E}_\beta | \Psi_j \rangle = \langle \Psi_i | \Psi_j \rangle = \delta_{ij}$. In particular this implies that the matrix element $c_{\alpha\beta}$ is either 0 in the case of (1) or 1 in the case of (2). Conditions (1) and (2) can be reformulated succinctly as: $\mathbf{E}_\alpha^\dagger \mathbf{E}_\beta \notin N(\mathcal{S}) - \mathcal{S}$.

The nine bit Shor code in Eq. (5) is a stabilizer code. The set of its stabilizers is generated by

$$\begin{array}{cccccc} Z_1Z_2 & Z_2Z_3 & Z_4Z_5 & Z_5Z_6 & Z_7Z_8 & Z_8Z_9 \\ \multicolumn{2}{c}{X_1X_2X_3X_4X_5X_6} & \multicolumn{2}{c}{X_1X_2X_3X_7X_8X_9} & \multicolumn{2}{c}{X_4X_5X_6X_7X_8X_9} \end{array}$$

Note that these generators correspond to the measurements to detect bit flip (the Z generators) and phase flip (the X generators) errors. Indeed, for instance a bit flip error on the first qubit anticommutes with Z_1Z_2.

The generators for the stabilizer of the Steane code in Eq. (9) on the 7 qubits are the following:

$$\begin{array}{cc} IIIZZZZ & IIIXXXX \\ IZZIIZZ & IXXIIXX \\ ZIZIZIZ & XIXIXIX. \end{array}$$

Note how the positions of the Z and X correspond to the parity check matrix H in Eq. (8). The fact that the code is self dual is seen in the symmetry between the Z and the X. Note that the change of basis from $|0\rangle$, $|1\rangle$ to the $|\pm\rangle$ basis, which is implemented by a conjugation with the Hadamard transform H on each bit, transforms all Z into X and vice versa ($HXH = Z$ and $HZH = X$).

The smallest possible quantum code to protect against single qubit errors, the 5-qubit code [LMPZ96], has the following (shift invariant) stabilizer

$$\begin{array}{c} XZZXI \\ IXZZX \\ XIXZZ \\ ZXIXZ. \end{array}$$

The stabilizer formalism allows to derive fault-tolerant computation in a convenient way. A key ingredient are encoded gates that transform encoded states, without decoding them, which would expose them to noise without protection. For universal quantum computation it is sufficient to show how to implement a universal *discrete* set of gates fault-tolerantly.

The key insight to fault-tolerant gates is that these encoded gates should only take encoded states to valid encoded states, without leaving the code-space. For an encoded gate G this means that after application of G to a state stabilized by all elements of $\mathcal{S}$ the resulting state must still be stabilized by $\mathcal{S}$ (see Eq. (17))

$$G|\Psi\rangle \in \mathcal{C} \quad \leftrightarrow \quad SG|\Psi\rangle = GS|\Psi\rangle \tag{18}$$

or in other words over the code-space G commutes with all elements of $\mathcal{S}$. In the case of QECCs and the Pauli group this means that G is in the *normalizer* $N(\mathcal{S})$ of $\mathcal{S}$.

The normalizer allows one to easily identify encoded logical operations on the code. It can be shown that for large classes of stabilizer codes a universal gate-set can be implemented either because the encoded gates are *transversal*, i.e. they affect only one qubit per block, or in connection with state-preparation of special states and measurement.

To fault-tolerantly *measure* on encoded states ancilla-state are employed in a procedure like the following[10]: Suppose we wish to measure the encoded qubit in the encoded computational basis. The ancilla is prepared in the state $|0_L\rangle$. Then we perform an *encoded* $CNOT$ from the encoded qubit to be measured to the ancilla. We then measure the ancilla state in the computational basis, which gives us a non-destructive measurement of the encoded qubit in the encoded computational basis which is tolerant of possible errors in the encoded qubit. To prevent possible uncontrolled error-propagation caused by an incorrectly prepared ancilla, we prepare multiple $|0_L\rangle$-ancillas and apply $CNOT$'s between the DFS state to be measured and each ancilla. Together with majority voting this provides a fault-tolerant method for measuring $\bar{Z}$ [Got97a].

The stabilizer formalism also provides an easy framework for fault-tolerant encoded state preparation and decoding, as it turns out that only transversal measurements in the Pauli-Z-basis are needed for both. For a detailed account of fault-tolerant computation with stabilizer codes see Gottesman [Got97b].

E Noise model for Decoherence-Free Subsystems

To derive the model of *collective* noise that applies to decoherence-free subsystems, we will work with the Hamiltonian picture (see App. B.1), following [ZR97]. We use the interaction Hamiltonian Eq. (13). *Collective decoherence* is the case where the coupling constants do not depend on the qubit, i.e. $g^{\alpha}_{i\mathbf{k}} = g^{\alpha}_{\mathbf{k}}$. This allows to rewrite the Hamiltonian in terms of the operators

$$S_\alpha = \sum_{i=1}^{n} \sigma^i_\alpha \tag{19}$$

as

$$H = \omega_0 S_z + \sum_{\mathbf{k}} \omega_{\mathbf{k}} b^\dagger_{\mathbf{k}} b_{\mathbf{k}} + \sum_{\alpha=\pm,z} S_\alpha \otimes \sum_{\mathbf{k}} g^\alpha_{\mathbf{k}} \tilde{\mathbf{B}}^\alpha_{\mathbf{k}} + h.c.$$

The crucial observation is now that if we start the system in a common eigenstate with the same eigenvalue of all the S_α with the bath in an eigenstate of H_B then the evolution will be completely *decoupled.* The condition for decoherence-free subspaces (DFS) is

$$S_\alpha|\Psi\rangle = c_\alpha|\Psi\rangle \quad \forall|\Psi\rangle \in DFS \tag{20}$$

Dynamical symmetry allows for unitary evolution of a subspace while the remaining part of the Hilbert space gets strongly entangled with the environment. This is true for arbitrary coupling strength. The form of noise where all three S_α, $\alpha \in \{x, y, z\}$, come into play is now called *strong collective decoherence*, if there is only coupling to one of the S_α the noise is called *weak collective decoherence.*

It is also possible to study collective noise in the Markovian picture (see App. B.3). We use Eq. (15), where $\mathbf{L}_D$ gives the non-unitary "coupling term[11]. The decoherence-free condition $\mathbf{L}_D[\rho] = 0$ implies that

$$\mathbf{F}_\alpha|\Psi\rangle = c_\alpha|\Psi\rangle \quad \forall|\Psi\rangle \in DFS$$

As before the decoherence-free states are common eigenstates of the operators $\mathbf{F}_\alpha$. In case of collective decoherence (symmetry of all the qubits), the $\mathbf{F}_\alpha$ are exactly the S_α of Eq. (19), and it is possible to show that the unitary term of the Master-equation does not affect the DFS to first order.

This line of reasoning can be generalized to decoherence-free *subsystems.* The S_α act on the system space. We can study the irreducible representations of this action and identify irreducible subspaces. For each irreducible representation there will be one or several irreducible subspaces on which the S_α act in the same way. The one dimensional subspaces will only get a phase factor, they correspond to the decoherence-free subspaces of Eq. (20). But the other subspaces are not

[10] This procedure may differ from case to case, here we only give an example for illustration.

[11] Note that the coupling of a system with an environment might also change the unitary part of the evolution of $-i[\mathbf{H}, \rho]$ by introducing an additional term to the system Hamiltonian, called the *Lamb-shift.*

lost for our purposes. Any irreducible subspace can be used to encode information, because the action of the noise operators S_α will keep the state within the subspace. Even though the state will change, its subspace will identify the encoded information. The number of irreducible subspaces corresponding to the same irreducible representation gives the number of different code words we can use.

The irreducible representations corresponding to the operators associated with strong collective decoherence (Eq. (19)) have been studied widely in physics, as they correspond to the *angular momentum* operators.

In the case of two qubits there is a single common eigenstate of the S_α, $\alpha \in \{x, y, z\}$, the singlet state

$$|\Phi\rangle = \frac{1}{\sqrt{2}}(|0\rangle|1\rangle - |1\rangle|0\rangle).$$

For three qubits there is no one-dimensional irreducible representation of the S_α, but there are two (identical) two-dimensional irreducibles subspaces into which we can encode as

$$|0\rangle_{code} = \left\{ \begin{array}{l} |\frac{1}{\sqrt{2}}(|010\rangle - |100\rangle) \\ |\frac{1}{\sqrt{2}}(|011\rangle - |101\rangle) \end{array} \right. \qquad |1\rangle_{code} = \left\{ \begin{array}{l} \frac{1}{\sqrt{6}}(-2|001\rangle + |010\rangle + |100\rangle) \\ \frac{1}{\sqrt{6}}(2|110\rangle - |101\rangle - |011\rangle) \end{array} \right.$$

Increasing the number of qubits the number of identical irreducible subspaces grows favorably, so that it is possible to encode at a good rate. For more references on this and the theory of fault-tolerant computation on such systems, see Sec. A.

References

[AGP05] P. Aliferis, D. Gottesman and J. Preskill, Quantum accuracy threshold for concatenated distance-3 codes, 2005. lanl-report quant-ph/0504218.

[AL87] R. Alicki and K. Lendi, *Quantum Dynamical Semigroups and Applications*, Number 286 in Lecture Notes in Physics. Springer-Verlag, Berlin, 1987.

[ALZ05] R. Alicki, D.A. Lidar and P. Zanardi, Are the assumptions of fault-tolerant quantum error correction internally consistent?, 2005. lanl-report quant-ph/0506201.

[A] D. Aharonov and M. Ben-Or, Fault-tolerant quantum computation with constant error, Proceedings of 29th Annual ACM Symposium on Theory of Computing (STOC),1997, 46, ACM, New York, NY.

[AvDK$^+$04] D. Aharonov, W. van Dam, J. Kempe, Z. Landau, S. Lloyd and O. Regev, Adiabatic quantum computation is equivalent to standard quantum computation, In *Proc. 45th Annual IEEE Symp. on Foundations of Computer Science (FOCS)*, 2004.

[Bac01] D. Bacon, Decoherence, control, and symmetry in quantum computers, 2001. Ph.D. thesis, University of California, Berkeley.

[BDSW96] C.H. Bennett, D.P. DiVincenzo, J.A. Smolin and W.K. Wootters, Mixed state entanglement and quantum error correction, *Phys. Rev. A* **54**, 3824 (1996).

[BKLW00] D. Bacon, J. Kempe, D.A. Lidar and K.B. Whaley, Universal fault-tolerant computation on decoherence free subspaces, *Phys. Rev. Lett.***85**, 1758–1761 (2000).

[BLW99] D. Bacon, D.A. Lidar and K.B. Whaley, Robustness of decoherence-free subspaces for quantum computation. *Phys. Rev. A*, 60:1944, 1999.

[BR01] H.J. Briegel and R. Raussendorf, A one-way quantum computer, *Phys. Rev. Lett.* **86** 5188 (2001).

[Car93] H. Carmichael, *An Open Systems Approach to Quantum Optics*, Number m18 in Lecture notes in physics. Springer-Verlag, Berlin, 1993.

[CFP02] A. Childs, E. Farhi and J. Preskill, Robustness of adiabatic quantum computation, *Phys. Rev. A* **65**, 012322 (2002).

[CL83] A.O Caldeira and A.J. Leggett, Quantum tunneling in a dissipative system, *Ann. of Phys.* **149** (2), 374–456 (1983).

[CRSS98] A.R. Calderbank, E.M. Rains, P.W. Shor and N.J.A. Sloane, Quantum error correction via codes over GF(4), *IEEE Trans. Inf. Th.* **44**, 1369 (1998).

[CS96] A.R. Calderbank and P.W. Shor, Good quantum error correcting codes exist. *Phys. Rev. A* **54**, 1098–1105 (1996).

[Dav76] E.B. Davies, *Quantum theory of open systems*, Academic Press, London, 1976.

[DBK$^+$00] D. P. DiVincenzo, D. Bacon, J. Kempe, G. Burkar and K. B. Whaley, Universal quantum computation with the exchange interaction, *Nature* **408**, 339 (2000).

[DG98] L.-M Duan and G.-C. Guo, Reducing decoherence in quantum-computer memory with all quantum bits coupling to the same environment, *Phys. Rev. A* **57**, 737 (1998).

[Die82] D. Dieks, Communication by electron-paramagnetic-resonance devices, *Phys. Lett.* **92A**, 271 (1982).

[DiV95] D. P. DiVincenzo, Two-bit gates are universal for quantum computation, *Phys. Rev. A* **51** (2), 1015–1022 (1995).

[FGG$^+$01] E. Farhi, J. Goldstone, S. Gutmann, J. Lapan, A. Lundgren and D. Preda, A quantum adiabatic evolution algorithm applied to random instances of an NP-complete problem, *Science* **5516**, 472–476 (2001).

[FKLW01] M.H. Freedman, A. Kitaev, M. Larsen and Z. Wang, Topological quantum computation, LANL preprint quant-ph/0101025, 2001.

[FKSS04] Jesse Fern, Julia Kempe, Slobodan Simic and Shankar Sastry, Fault-tolerant quantum computation - a dynamical systems approach, 2004. quant-ph/0409084.

[Got97a] D. Gottesman, Class of quantum error-correcting codes saturating the quantum hamming bound, *Phys. Rev. A* **54**, 1862 (1997).

[Got97b] D. Gottesman, *Stabilizer codes and quantum error correction*, PhD thesis, California Institute of Technology, Pasadena, CA, 1997.

[Got97c] D. Gottesman, Theory of fault-tolerant quantum computation, *Phys. Rev. A* **57**, 127 (1997).

[Got05] D. Gottesman, Quantum error correction and fault-tolerance, 2005.

[Kal05] G. Kalai, Thoughts on noise and quantum computation, 2005. lanl-report quant-ph/0508095.

[KBLW01] J. Kempe, D. Bacon, D.A. Lidar and K.B. Whaley, Theory of decoherence-free fault-tolerant universal quantum compuation, *Phys. Rev. A* **63**, 042307 (2001).

[Kem01] J. Kempe, *Universal Noiseless Computation: Mathematical Theory and Applications*, PhD thesis, University of California, Berkeley, 2001.

[Kit97] A.Yu. Kitaev, Quantum computations: Algorithms and error corrections, *Russian Math. Surveys* **52**, 1191–1249 (1997).

[Kit03] A.Y. Kitaev, Fault-tolerant quantum computation by anyons, *Ann. of Phys.* **303**, 2–30 (2003).

[KL96] E. Knill and R. Laflamme, Concatenated quantum codes, 1996. LANL preprint quant-ph/9608012.

[KL97] E. Knill and R. Laflamme, Theory of quantum error-correcting codes, *Phys. Rev. A* **55**, 900 (1997).

[KLV00] E. Knill, R. Laflamme and L. Viola, Theory of quantum error correction for general noise, *Phys. Rev. Lett.* **84**, 2525 (2000).

[KLZ96] E. Knill, R. Laflamme and W. H. Zurek, Accuracy threshold for quantum computation, Technical report, Quantum Physics e-Print archive, 1996. `http://xxx.lanl.gov/abs/quant-ph/9611025`.

[KLZ98] E. Knill, R. Laflamme and W. Zurek, Resilient quantum computation, *Science* **279**, 342–345 (1998).

[Kni04] E. Knill, Fault-tolerant postselected quantum computation: Threshold analysis, 2004.

[Kni05] E. Knill, Quantum computing with realistically noisy devices, *Nature* **434**, 39–44 (2005).

[Kra83] K. Kraus, *States, Effects and Operations*, Fundamental Notions of Quantum Theory. Academic, Berlin, 1983.

[KSV02] A.Y. Kitaev, A.H. Shen and M.N. Vyalyi, *Classical and Quantum Computation*, Number 47 in Graduate Series in Mathematics. AMS, Providence, RI, 2002.

[Lan95] R. Landauer, Is quantum mechanics useful? *Phil. Tran, Roy. Soc. Lond.* **353**, 367 (1995).

[LBW99] D.A. Lidar, D. Bacon and K.B. Whaley, Concatenating decoherence free subspaces with quantum error correcting codes, *Phys. Rev. Lett.* **82**, 4556 (1999).

[LCD$^+$87] A.J. Leggett, S. Charkavarty, A.T. Dorsey, M.P.A. Fisher, A. Garg and W. Zwerger, Dynamics of the dissipative two-state system, *Rev. Mod. Phys.* **59** (1), 1–85 (1987).

[LCW98] D.A. Lidar, I.L. Chuang and K.B. Whaley, Decoherence free subspaces for quantum computation, *Phys. Rev. Lett.* **81**, 2594 (1998).

[Lin76] G. Lindblad, On the generators of quantum dynamical semigroups, *Commun. Math. Phys.* **48**, 119 (1976).

[LMPZ96] R. Laflamme, C. Miquel, J.P. Paz and W.H. Zurek, Perfect quantum error correction code, *Phys. Rev. Lett.* **77**, 198 (1996).

[LW03] D.A. Lidar and K.B. Whaley, *Lecture Notes in Physics*, volume 622, chapter Decoherence-Free Subspaces and Subsystems, pages 83–120. Sringer, Berlin, 2003.

[NC00] M.A. Nielsen and I.L. Chuang, *Quantum Computation and Quantum Information.* Cambridge University Press, Cambridge, UK, 2000.

[ND05] M.A. Nielsen and C.M. Dawson, Fault-tolerant quantum computation with cluster states, *Phys. Rev. A* **71**, 042323 (2005).

[Neu56] J. von Neumann, Probabilistic logics and the synthesis of reliable organisms from unreliable components, In *Automata Studies*, pages 329–378, Princeton, NJ, 1956. Princeton University Press.

[Pre98a] J. Preskill, Quantum information and computation, lecture notes, `http://www.theory.caltech.edu/people/preskill/ph229/`, 1998.

[Pre98b] J. Preskill, Reliable quantum computers, *Proc. R. Soc. A* **454**, 385–410 (1998).

[Pre99] J. Preskill, Fault-tolerant quantum computation, In T.P. Spiller H.K. Lo, S. Popescu, editor, *Introduction to quantum computation*, page 213, Singapore, 1999. World Scientific.

[PSE96] G. Palma, K. Suominen and A. Ekert, Quantum computers and dissipation, *Proc. Roy. Soc. London Ser. A* **452**, 567 (1996).

[Rei05] B. Reichardt, Threshold for the distance three steane quantum code, 2005. lanl-report quant-ph/0509203.

[SBFH05] S. H. Simon, N.E. Bonesteel, M.H. Freedman and N. Petrovicand L. Hormozi, Topological quantum computing with only one mobile quasiparticle, 2005. lanl-archive quant-ph/0509175.

[Sho94] P.W. Shor, Algorithms for quantum computation: Discrete log and factoring. In *Proceedings of the 35th Annual Symposium on the Foundations of Computer Science*, pages 124–134, Los Alamitos, CA, 1994. IEEE Computer Society.

[Sho95] P.W. Shor, Scheme for reducing decoherence in quantum memory, *Phys. Rev. A* **52**, 2493–2496 (1995).

[Sho96] P.W. Shor, Fault-tolerant quantum computation, In *Proceedings of the 37th Symposium on Foundations of Computing*, pages 56–65, Los Alamitos, CA, 1996. IEEE Computer Society Press.

[Ste96a] A. Steane, Multiple particle interference and quantum error correction, *Proc. Roy. Soc. London* **452**, 2551–2577 (1996).

[Ste96b] A.M. Steane, Error correcting codes in quantum theory, *Phys. Rev. Lett.* **77**, 793 (1996).

[Ste97] A.M. Steane, Active stabilisation, quantum computation and quantum state synthesis, *Phys. Rev. Lett.* **78**, 2252–2255 (1997).

[Ste99] A.M. Steane, Quantum error correction, In S. Popescu H.K. Lo and T.P. Spiller, editors, *Introduction to Quantum Computation and Information*, page 184. World Scientific, Singapore, 1999.

[Ste01] A.M. Steane, *Decoherence and its implications in quantum computation and information transfer*, chapter Quantum Computing and Error Correction, pages 284–298. IOS Press, Amsterdam, 2001.

[Ste03] A.M. Steane, Overhead and noise threshold of fault-tolerant quantum error correction, *Phys. Rev. A* **68**, 042322 (2003).

[TB05] B. M. Terhal and G. Burkard, Fault-tolerant quantum computation for local nonmarkovian noise, *Phys. Rev. A* **71**, 012336 (2005).

[Unr95] W.G. Unruh, Maintaining coherence in quantum computers, *Phys. Rev. A* **51**, 992–997 (1995).

[VKL99] L. Viola, E. Knill and S. Lloyd, Dynamical decoupling of open quantum systems, *Phys. Rev. Lett.* **82**, 2417 (1999).

[WZ82] W. Wootters and W. Zurek, A single quantum cannot be cloned, *Nature* **299**, 802 (1982).

[Zan97] P. Zanardi, Dissipative dynamics in a quantum register, *Phys. Rev. A* **56**, 4445 (1997).

[Zan98] P. Zanardi, Dissipation and decoherence in a quantum register, *Phys. Rev. A* **57**, 3276 (1998).

[ZR97] P. Zanardi and M. Rasetti, Error avoiding quantum codes, *Mod. Phys. Lett. B* **11**, 1085 (1997).

[Zur91] W.H. Zurek, Decoherence and the transition from quantum to classical, *Phys. Today* Oktober, 36–44 (1991).